FEKO
User's Manual

Suite 4.0
(FEKO 45.154, PREFEKO 26.1-53 and later)

December 2002

Copyright 1998 – 2002: **EM Software & Systems-S.A. (Pty) Ltd**
32 Techno lane, Technopark, Stellenbosch, 7600, South Africa
Tel: +27-21-880-1880, Fax: +27-21-880-1936
E-Mail: feko@emss.co.za
WWW: http://www.feko.info



Contents

1 Introduction **1-1**

2 General comments **2-1**

2.1 Structure of the input file . . . 2-1
2.2 Summary of the files . . . 2-2
2.3 Entering the geometry . . . 2-4
2.4 Utilisation of symmetry . . . 2-7
2.4.1 Geometric symmetry . . . 2-7
2.4.2 Electric symmetry . . . 2-7
2.4.3 Magnetic symmetry . . . 2-8
2.4.4 Example of the application of symmetry . . . 2-8
2.4.5 Special enforcement of symmetry: Even – odd method . . . 2-10
2.5 Dielectric solids . . . 2-11
2.6 Dynamic memory management . . . 2-12
2.6.1 Setting maxalloc for out-of-core solutions . . . 2-12
2.6.2 Other variables that are under user control . . . 2-13
2.6.3 Variables that are automatically set correctly . . . 2-13
2.7 Environment variables . . . 2-15
2.8 Checking the validity of the results . . . 2-18

3 The program WinFEKO **3-1**

3.1 Hardware and software requirements for WinFEKO . . . 3-1
3.1.1 Operating systems . . . 3-1
3.1.2 Memory requirements . . . 3-1
3.1.3 Display settings . . . 3-1
3.1.4 Graphics card . . . 3-2
3.2 Running WinFEKO . . . 3-2
3.3 Hot-keys in WinFEKO . . . 3-2

3.4 Toolbars in WinFEKO . . . 3-3
3.4.1 FILE control toolbar . . . 3-3
3.4.2 FEKO control toolbar . . . 3-4
3.4.3 DISPLAY OPTIONS control toolbar . . . 3-4
3.4.4 RESULTS control toolbar . . . 3-5
3.4.5 RENDER control toolbar . . . 3-5
3.5 Main menu structure . . . 3-6
3.5.1 File menu . . . 3-6
3.5.2 Preprocessing menu . . . 3-10
3.5.3 Solve menu . . . 3-11
3.5.4 Running the parallel version of FEKO . . . 3-12
3.5.5 Results menu . . . 3-13
3.5.6 Display menu . . . 3-23
3.5.7 Tools menu . . . 3-29
3.5.8 Options menu . . . 3-31
3.5.9 Help menu . . . 3-33

4 The editor EditFEKO 4-1
4.1 General . . . 4-1
4.1.1 File menu . . . 4-1
4.1.2 Options menu . . . 4-2
4.1.3 Window menu . . . 4-2
4.1.4 Help menu . . . 4-2
4.2 PREFEKO mode . . . 4-2
4.2.1 Generating input cards . . . 4-2
4.2.2 Parameter suggestion . . . 4-4
4.2.3 Variable editor . . . 4-5
4.2.4 Edit menu . . . 4-5
4.2.5 Search menu . . . 4-6
4.2.6 Run menu . . . 4-6
4.3 OPTFEKO mode . . . 4-6
4.4 Important keystrokes . . . 4-7

5 The program GraphFEKO **5-1**

5.1 Running GraphFEKO 5-1

5.2 Toolbars in GraphFEKO 5-1

5.2.1 FILE control toolbar 5-1

5.2.2 Data extraction toolbar 5-2

5.3 Main menu structure 5-2

5.3.1 File menu 5-2

5.3.2 Import menu 5-5

5.3.3 Edit menu 5-16

5.3.4 Tools menu — Line arithmetics 5-19

5.3.5 Tools menu — Unwrap phase 5-20

5.3.6 Window menu 5-20

5.3.7 Help menu 5-21

6 The preprocessor PREFEKO **6-1**

6.1 Description 6-1

6.2 Running PREFEKO 6-1

6.3 Symbolic variables 6-1

6.4 FOR/NEXT loops 6-6

6.5 IF/ELSE/ENDIF constructs 6-8

6.6 Symbolic node names 6-9

6.7 PRINT and EXIT commands 6-9

6.8 Copyright to Voronoi 6-10

7 The program FEKO **7-1**

7.1 Introduction 7-1

7.2 Running FEKO 7-1

7.2.1 Running the sequential version 7-1

7.2.2 Running the parallel version 7-2

8 Description of the geometry cards **8-1**

8.1 Overview of the geometry cards 8-1

8.2 Alphabetical description of the geometry cards 8-3

8.2.1 ** Card 8-3

8.2.2 BL Card 8-4

8.2.3 BP Card 8-6

8.2.4 BQ Card 8-8

8.2.5 BT Card 8-11

8.2.6 CB Card 8-13

8.2.7 CL Card 8-14

8.2.8 CN Card 8-17

8.2.9 DK Card 8-18

8.2.10 DP Card 8-20

8.2.11 DZ Card 8-21

8.2.12 EG Card 8-23

8.2.13 EL Card 8-26

8.2.14 FO Card 8-28

8.2.15 HE Card 8-29

8.2.16 IN Card 8-31

8.2.17 IP Card 8-47

8.2.18 KA Card 8-49

8.2.19 KK Card 8-50

8.2.20 KL Card 8-55

8.2.21 KR Card 8-56

8.2.22 KU Card 8-60

8.2.23 LA Card 8-62

8.2.24 ME Card 8-63

8.2.25 NU Card 8-67

8.2.26 PB Card 8-71

8.2.27 PH Card 8-73
8.2.28 PM Card 8-78
8.2.29 PO Card 8-81
8.2.30 PY Card 8-84
8.2.31 QU Card 8-86
8.2.32 SF Card 8-88
8.2.33 SU Card 8-90
8.2.34 SY Card 8-91
8.2.35 TG Card 8-92
8.2.36 TO Card 8-95
8.2.37 TP Card 8-98
8.2.38 UT Card 8-99
8.2.39 UZ Card 8-101
8.2.40 VS Card 8-103
8.2.41 WG Card 8-106
8.2.42 ZY Card 8-108

9 Description of the control cards **9-1**

9.1 Overview of control cards and remarks on execution sequence 9-1
9.2 Detailed description of the control cards 9-4
9.2.1 ** Card 9-4
9.2.2 Ax Cards 9-5
9.2.3 A0 Card 9-8
9.2.4 A1 Card 9-11
9.2.5 A2 Card 9-12
9.2.6 A3 Card 9-13
9.2.7 A4 Card 9-14
9.2.8 A5 Card 9-16
9.2.9 A6 Card 9-17

9.2.10 A7 Card 9-19
9.2.11 AC Card 9-20
9.2.12 AE Card 9-23
9.2.13 AI Card 9-25
9.2.14 AP Card 9-27
9.2.15 AR Card 9-34
9.2.16 AV Card 9-39
9.2.17 BO Card 9-41
9.2.18 CG Card 9-43
9.2.19 CM Card 9-46
9.2.20 CO Card 9-47
9.2.21 DA Card 9-50
9.2.22 DI Card 9-52
9.2.23 EN Card 9-53
9.2.24 FE Card 9-54
9.2.25 FF Card 9-63
9.2.26 FR Card 9-65
9.2.27 GF Card 9-67
9.2.28 L4 Card 9-74
9.2.29 LD Card 9-75
9.2.30 LE Card 9-76
9.2.31 LP Card 9-77
9.2.32 LS Card 9-78
9.2.33 LZ Card 9-79
9.2.34 OF Card 9-80
9.2.35 OS Card 9-81
9.2.36 PS Card 9-83
9.2.37 PW Card 9-85
9.2.38 SK Card 9-89
9.2.39 SP Card 9-93
9.2.40 TL Card 9-94

10 The optimiser OPTFEKO **10-1**
10.1 Description . . . 10-1
10.2 The *.pre input file . . . 10-1
10.3 The *.opt input file . . . 10-1
10.3.1 Definition of optimisation parameters . . . 10-2
10.3.2 Definition of the penalty function . . . 10-2
10.3.3 Definition of the optimisation process . . . 10-3
10.3.4 Defining the aim function . . . 10-5
10.4 Running OPTFEKO . . . 10-15
10.5 An example using OPTFEKO . . . 10-15

11 The program TIMEFEKO **11-1**
11.1 Description . . . 11-1
11.2 The *.pre input file . . . 11-1
11.3 The *.tim input file . . . 11-2
11.3.1 Defining the pulse form . . . 11-2
11.3.2 Definition of the frequency block . . . 11-6
11.3.3 Definitions of the normalisation . . . 11-7
11.3.4 Definition of the excitation output . . . 11-7
11.3.5 Definition of a time point . . . 11-7
11.4 Running TIMEFEKO . . . 11-8
11.4.1 TIMEFEKO output . . . 11-8
11.5 A TIMEFEKO example . . . 11-8

12 The program LFFEKO **12-1**
12.1 Description . . . 12-1
12.2 The *.pre input file . . . 12-1
12.3 The *.geo input file . . . 12-1
12.3.1 CUTEDGE command . . . 12-2
12.3.2 DELLOOP command . . . 12-2
12.3.3 INSLOOP command . . . 12-2
12.3.4 END command . . . 12-3
12.4 Example . . . 12-3

13 The program ADAPTFEKO **13-1**

13.1 Description 13-1

13.2 Running ADAPTFEKO 13-1

13.3 The *.pre input file 13-1

13.4 ADAPTFEKO example 13-2

14 Description of the output file of FEKO **14-1**

14.1 Geometric data 14-1

14.2 Excitation 14-6

14.3 Currents and charges 14-7

14.4 Finite conductivity 14-11

14.5 Near field 14-12

14.6 Far fields 14-13

14.7 S-parameters 14-15

14.8 Computation time 14-16

Index **I-1**

1 Introduction

The name FEKO is an abbreviation derived from the German phrase **FE***ldberechnung bei* **K***örpern mit beliebiger* **O***berfläche.* (*Field computations involving bodies of arbitrary shape.*) As the name suggests FEKO can be used for various types of electromagnetic field analyses involving objects of arbitrary shapes.

The program FEKO is based on the *Method of Moments* (MoM). Electromagnetic fields are obtained by first calculating the electric surface currents on conducting surfaces and equivalent electric and magnetic surface currents on the surface of a dielectric solid. The currents are calculated using a linear combination of basis functions, where the coefficients are obtained by solving a system of linear equations. Once the current distribution is known, further parameters can be obtained e.g. the near field, the far field, radar cross sections, directivity or the input impedance of antennas.

Electrically large problems are usually solved with either the Physical Optics (PO) approximation and its extensions or the Uniform Theory of Diffraction (UTD). In FEKO these formulations are hybridised with the MoM at the level of the interaction matrix[1]. This is a major step in addressing the problem of solving electromagnetic problems where the object under consideration is too large (in terms of wavelengths) to solve with the MoM but too small to apply only the asymptotic UTD approximation with high accuracy. With the hybrid MoM/PO or hybrid MoM/UTD techniques, critical regions of the structure can be considered using the MoM and the remaining regions (usually larger, flat or curved metallic surfaces) using the PO approximation or UTD.

Only time domain harmonic sources are supported in the current version, and consequently calculation is done in the frequency domain. FEKO uses the $e^{j\omega t}$ time convention. Different sources are available including an incident plane wave, various voltage gap formulations (between wire segments), and a magnetic ring current (TEM-Frill, with which a coaxial feed can be modelled).

WinFEKO is the main user interface module and is used to control the solution of a problem. The geometry is defined in terms of high level commands in an input file (`*.pre`) which also sets the solution parameters. The customised text-editor EditFEKO assists the user in creating and editing the input file. The preprocessor/mesher PREFEKO processes this file and prepares the input file (`*.fek`) for the program FEKO which is the actual field calculation code. PREFEKO enables the user to create complex geometries with a single command, for example coils consisting of wire segments, or flat, cylindrical or spherical plates consisting of triangles.

The geometry can also be created in the 3D CAD/mesher tool FEMAP. This method is preferred for very complex geometries or for importing data from other CAD packages. The meshed geometry is then exported from FEMAP and imported by PREFEKO.

[1] The UTD implementation is currently limited to an arbitrary number of perfectly conducting, flat, polygonal surfaces or one perfectly conducting circular cylinder.

Note that PREFEKO also supports the direct import of *meshed* geometry in NASTRAN and AutoCAD `*.dxf` formats. For simpler geometries, the commands processed by PREFEKO are much more efficient. In addition to controlling the solution process, WinFEKO is used to visualise the geometry created by FEMAP/PREFEKO.

The output file (`*.out`) of FEKO contains all the solution information. The resulting fields and/or currents can be displayed in 3D with WinFEKO or as 2D graphs using GraphFEKO.

Note that the preprocessor PREFEKO and the field computation program FEKO are available on PC's and a wide variety of workstations. The programs WinFEKO, EditFEKO and GraphFEKO are available on PC's only. All pre- and post processing must thus be performed on a PC, while the actual computationally intensive field calculations can be performed on the PC itself, or on a workstation or parallel cluster as required.

First time users are advised to work through the *Getting Started* manual (located in the `doc` directory under the FEKO home directory — the directory where FEKO has been installed). It gives a basic introduction to FEKO and the different FEKO modules. It is also recommended that new users read the "General Comments" (Chapter 2) of this *FEKO User's Manual* carefully.

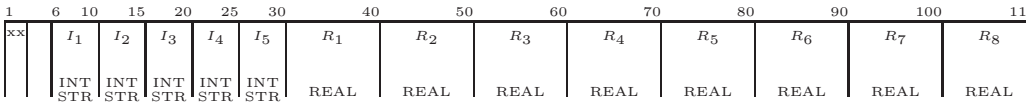
Various simple FEKO examples that show the application of the various cards are are discussed in the *Examples Guide.*

Changes in this manual with respect to the previous manual of March 2002 (Suite 3.2) are indicated as follows:
Sections that has changed from those in the previous version of the manual.
Sections that were newly added to this version of the manual.

2 General comments

2.1 Structure of the input file

The program is card driven (similar to the thin wire MoM code NEC). An input file consists of various cards (high level commands) which can be categorised into geometry cards and control cards. The geometry and the field parameters to be calculated are specified using these cards. Certain geometric cards are interpreted and filtered out by the preprocessor PREFEKO and translated into other cards. The basic form of the input cards is



The upper numbers indicate the columns. The name field in columns 1 and 2 specifies the type of the card. This is followed by five integer parameters I_1 to I_5 (when read with PREFEKO these input fields may also contain strings such as node names) containing five digits each, and eight real parameters R_1 to R_8 containing ten digits each.

When entering the data into the cards the user must ensure that all parameters are entered in the correct columns. The editor EditFEKO, assists the user to place the parameters in the correct columns through the use of dialog boxes. All parameters are in SI units, e.g. lengths are in metres, potential in volts, etc. (Note the angles are in degrees.) FEKO includes various scaling options (see the SF, TG and IN cards) such that dimensions may be entered in different units and scaled to metres.

The first line of the input file must contain the filename (without extension). The full filename, including the path, can be up to 254 characters long. This specifies the filename to which different extensions are added for various output files (see section 2.2). Alternatively the first line can be a comment (** card) or empty line.

The principal structure of the input file is shown below:

`**`	*Comments at the start of the input file*
...	*Cards that define the geometry*
`EG`	*End of the geometry*
...	*control cards that define the excitation and indicate which field quantities are to be calculated*
`EN`	*End of the input file*

Chapter 8 gives an overview of the geometric cards with detailed descriptions of the individual cards. Similarly Chapter 9 gives an overview and descriptions of the control cards.

2.2 Summary of the files

The table below gives an overview of the different files and their respective functions. STDOUT is the standard output, usually the screen. A * is used to symbolically indicate the filename.

Filename	Description
STDOUT	Usually the screen. This is where comments such as progress, warnings and errors are sent.
*._14	Page (temporary storage) file for the matrix in the sequential version of FEKO with out-of-core solution.
*._15	Page (temporary storage) file for the matrix in the sequential version of FEKO with out-of-core solution.
*._16	Page (temporary storage) file for the matrix in the sequential version of FEKO with out-of-core solution.
*._20	Page (temporary storage) file for the coupling coefficients of the MoM/PO hybrid method during an out-of-core solution.
*._21	Page (temporary storage) file for the coupling coefficients of the MoM/PO hybrid method during an out-of-core solution.
*.afo	Continuous frequency results created by ADAPTFEKO.
*.aus	Output file of TIMEFEKO.
*.bof	Binary version of the output file which is used for post processing.
*.cgm	Contains the size of the residue that results from the iterative algorithm which solves the matrix equation and the number of iterations. This file is only generated on request by a DA card (section 9.2.21).
*.dbg	When using the UTD, it is possible to request an optional output file containing a large amount of additional data (and may therefore be very large), see the UT card.
*.dxf	AutoCAD geometry file which can be imported with the IN card. (Only lines and meshed polygons are imported, see section 8.2.16.)
*.edg	Geometric data is taken from the *.fek input file, where for example common edges between triangles are found. This reprocessed information of the geometry is saved in the *.edg file. (EG card, section 8.2.12)
*.efe	File containing the electric field strengths. Contains both the position and the complex components of the electric field strength vectors. This file is only generated on request by a DA card (section 9.2.21).
*.fek	Output file from PREFEKO — serves as the input file for FEKO.
*.ffe	File containing the far field data. This file is only generated on request by a DA card (section 9.2.21).
*.geo	In addition to the *.pre file, a *.geo file is required with the special, low frequency program — LFFEKO (low frequency FEKO, 12).
*.gfe	Interpolation table of the electric field strengths for the Green's function of a layered sphere.
*.gfh	Interpolation table of the magnetic field strengths for the Green's function of a layered sphere.

`*.hfe`	File containing the magnetic field strengths. Contains both the position and the complex components of the magnetic field strength vectors. This file is only generated on request by a DA card (section 9.2.21).
`*.isd`	Data file containing the field distribution calculated by FEKO for coupling with CableMod.
`*.log`	Log file created by OPTFEKO.
`*.mat`	File in which the matrix elements of the linear equation system, are stored (only generated on request of a PS card (section 9.2.36).
`*.mod`	FEMAP model file.
`*.nas`	NASTRAN geometry file which can be imported with the IN card.
`*.neu`	Geometrical data file which is exported by the program FEMAP.
`*.opt`	Input file for the program OPTFEKO.
`*.ofc`	Paging files for the array elements used with sequential and parallel out-of-core solution. (To avoid the 2 GByte file size limit; or on parallel systems with a distributed file system, several files may be used. These are distinguished by adding numbers to the filename.)
`*.os`	File containing the surface currents and the currents in the segments. Data is in the form of position and the complex components of the current density vectors. This file is only generated on request by a DA card (section 9.2.21).
`*.out`	Output file from FEKO in which the results of all the calculations and messages can be found.
`*.pre`	Input file for PREFEKO.
`*.ray`	For the UTD an optional ray file can be requested allowing the ray paths to be visualised in WinFEKO.
`*.rhs`	File containing the right hand side vector in the system of linear equations.
`*.rsd`	File for coupling of FEKO with CableMod. It is usually created by CableMod, but can also be created by FEKO if requested with the OS card (field calculation along lines).
`*.snp`	Touchstone format S-parameter file as created by the DA card. The `n` refers to the number of ports (see section 9.2.21).
`*.str`	File in which the currents (the coefficients of the basis functions) are stored for reuse (generated on request from a PS card, section 9.2.36).
`*.tim`	Input file for the program TIMEFEKO.
`*.vis`	When multiple reflections are used with the PO formulation FEKO determines which basis functions have line of sight visibility. Since this calculation may require significant run time, this information can be saved to a `*.vis` file for reuse.
`*.wfg`	Figure created and saved with GraphFEKO.
`*.wfp`	WinFEKO project file.

The files `*.efe`, `*.hfe`, `*.ffe` and `*.os` are redundant. All the information in these files is also available in the `*.out` file. The format of these redundant files lends itself more readily to further processing.

2.3 Entering the geometry

Conducting surfaces are subdivided into triangles, and wires into segments. For dielectrics, there are a number of possibilities — see the discussion "Dielectric Solids" in section 2.5. Using the surface current method the surface of the dielectric solid is also subdivided into triangles, whereas with the volume current method, the solid is subdivided into cuboids.

Given below are a number of definitions that are used frequently in this manual:

Segment: A short section of a wire (short in comparison with the wavelength).

Node: The point where two segments are joined is called a node. One basis function is assigned to each node.

Edge: The common line between two adjacent triangles. If the surface is a metal, then one basis function is assigned to each edge. If the surface is a dielectric, then two basis functions are assigned for the current density, one for the equivalent electric current density and one for the equivalent magnetic current density.

Connection point: A connection point is where a segment is joined to a triangle. The end of the segment is connected to the vertex of the triangle. A basis function is assigned to each connection point.

Cuboid: A volume element used to model dielectric and magnetic solids according to the volume current method. It has 90 degree corners similar to a cube, but does not have to have equal side lengths.

Polygon: A planar surface element with straight edge boundaries.

The segmentation (or meshing) is performed automatically by the program PREFEKO, but there are some rules that have to be adhered to.

Wires can only be connected at the end points, of the respective segments. They are not allowed to overlap. An example is shown in figure 2-1. The wires AB and CD are not allowed to be entered in this way. AB will be subdivided into segments so that point C is not recognised as an ohmic connection. This is shown, on the left side, in figure 2-2. The line AB is divided into four segments and the point C is not on a node. To resolve this problem three wires have to be defined AC, CB and CD. Then there will be an ohmic connection at point C, as shown in figure 2-2 on the right.

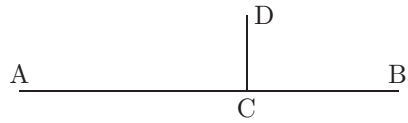


Figure 2-1: Example of a wire structure



Figure 2-2: Incorrect (left) and correct (right) subdivision into segments

A similar rule has to be followed when surfaces are to be subdivided into triangles. If the surface in figure 2-3 has to be meshed, there are a number of possibilities. The surface can be subdivided into the rectangles ABFG and CDEF. This can result in the mesh shown in figure 2-4 (on the left-hand side). In this case, there is no ohmic connection at the line BF, because the triangles' vertices are not connected (in the sense that the triangles do not have common edges).

The correct subdivision of the surface shown in figure 2-3 is shown in figure 2-4 (in the middle). Here the rectangles ABFG, CDHB and BHEF were used. Another subdivision is possible using "rectangles" ABEG, and BCDE (figure 2-4, right side). BE is now a common edge and the surface will be meshed correctly.

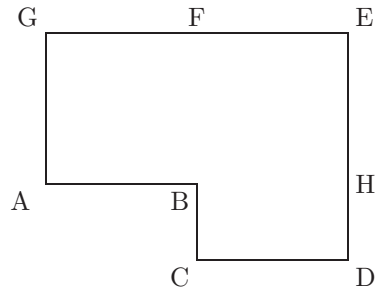


Figure 2-3: Example for a surface

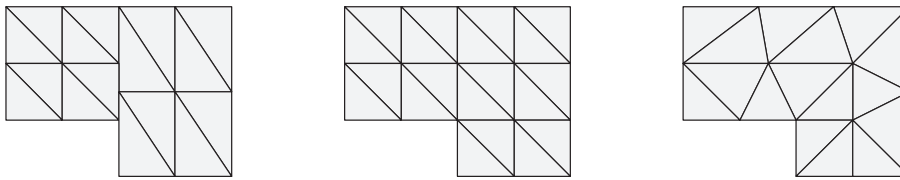


Figure 2-4: Incorrect (left) and correct (middle and right) subdivision into triangles

Care has to be taken to ensure that when two surfaces touch, the common edge is part of both surfaces. In figure 2-3 the edge BF is such an edge. That is why the right part of the surface is not allowed to be defined as the rectangle CDEF, because BF is not an edge of this rectangle.

A connection point between a segment and one or more triangles is only recognised when the beginning or the end of the segment lies on the vertex or vertices of the triangles. In figure 2-5 the left side is an incorrect and the right side a correct connection (here the segment is connected to six triangles).

When curved structures (circles, spheres) are modelled, a finer mesh may be used. When connecting curved edges, the same maximum edge length has to be used for both edges. See the example in figure 2-6.

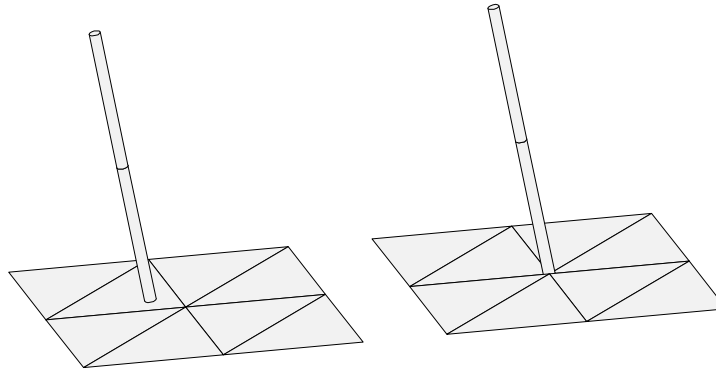


Figure 2-5: Incorrect (left) and correct (right) connection between a segment and triangles

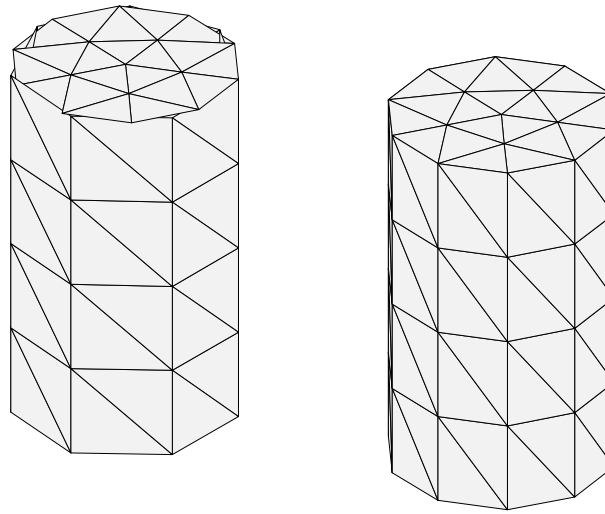


Figure 2-6: Incorrect (left) and correct (right) connection between curved edges

2.4 Utilisation of symmetry

It is possible to reduce the calculation time and memory usage if symmetry is utilised. This can be done by using the SY card (section 8.2.34).

Three coordinate planes, $x = 0$ (yz plane), $y = 0$ (xz plane) and $z = 0$ (xy plane) may be defined as planes of symmetry. There are three different types of symmetry. They are described below.

2.4.1 Geometric symmetry

With this type of symmetry the geometry of the modelled solid or part of the solid is symmetric about one or more coordinate planes. *The interaction between any two basis functions must be the same as that between their symmetrical counterparts. Everything which affects this must be symmetrical, i.e. loading, losses, Green's functions, etc.* The source, however, is not symmetric, thus a symmetric current distribution does not exist. This asymmetric current distribution leads to asymmetric electric and magnetic fields.

The body of a truck with an antenna placed at the front left hand side, will be used as an example. In the input file, half of the body is constructed (either the left or right side). The other half is then created with the SY command. Finally the antenna is placed in the correct position on one side.

A rectangular metallic plate, illuminated by an electromagnetic wave from a direction outside the principle planes, is another example. In this case a quarter of the plate is constructed and the rest is created using the SY card (section 8.2.34) with geometric mirroring around two coordinate planes.

Geometric symmetry does not reduce the number of unknown coefficients in the current basis functions. Therefore there is no reduction in memory usage. There is, however, a reduction in computation time when the matrix elements are determined.

2.4.2 Electric symmetry

An electric symmetry plane is a plane which can be replaced by an ideal electrically conducting wall without changing the field distribution.

In figure 2-7 an electric symmetry plane is displayed. The electric $\vec{E}$ field's tangential component disappears and the magnetic $\vec{H}$ field's normal component disappears. The electric current density $\vec{J}$ is anti-symmetric and the magnetic current density $\vec{M}$ is symmetric.

As with geometric symmetry, less computational time is required to calculate the matrix elements. The number of unknown coefficients of the current basis functions are reduced, leading to linear equation system of a lower order. This leads to a further reduction in the computation time and requires less memory due to the reduction in matrix elements.

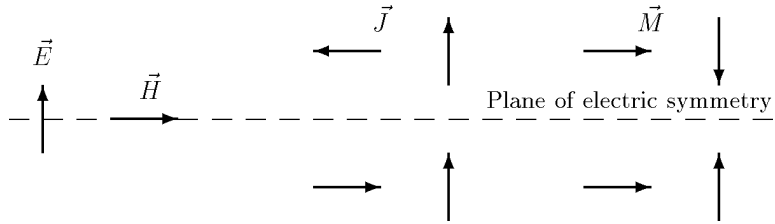


Figure 2-7: Electric symmetry plane

2.4.3 Magnetic symmetry

A magnetic symmetry plane is a plane which can be replaced by an ideal magnetically conducting wall without changing the field distribution.

In figure 2-8 a magnetic symmetry plane is displayed.

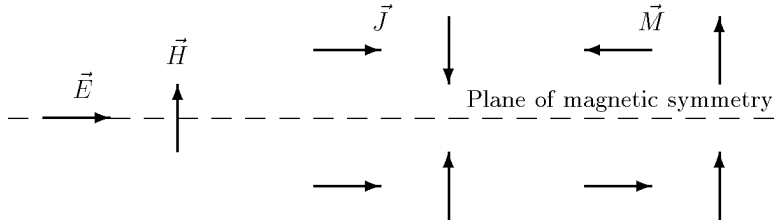


Figure 2-8: Magnetic symmetry plane

The electric $\vec{E}$ field's normal component and the magnetic $\vec{H}$ field's tangential disappears. The electric current density $\vec{J}$ is symmetric and the magnetic current density $\vec{M}$ is asymmetric.

When using magnetic symmetry there is a reduction in the computational time when determining the matrix elements. The order of the matrix equation is reduced, which leads to a further reduction in the computational time and reduces the amount of memory needed to determine the matrix elements.

2.4.4 Example of the application of symmetry

In figure 2-9 a dielectric sphere is shown with a linear polarised incident electromagnetic field. The full description of the problem is given in `example_04` in the *Examples Guide*. Only the use of symmetry is described here.

The plane $z = 0$ (xy plane) is a plane of geometric symmetry, because the excitation does not have any symmetry in this plane. The plane $x = 0$ (yz plane) is a plane of electric symmetry, because the electric field is perpendicular to this plane and the magnetic

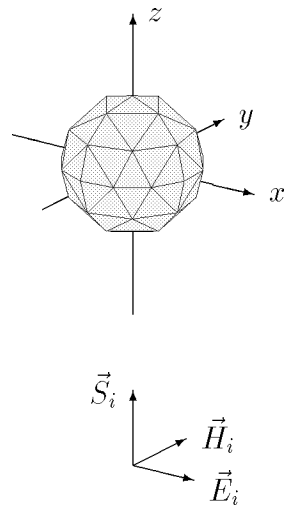


Figure 2-9: Dielectric sphere with incident field

field has a tangential component only. Similarly the electric field only has a tangential component in the $y = 0$ plane and the magnetic field is perpendicular. This is thus a plane of magnetic symmetry.

To indicate the reduction in time and resources through the use of symmetry, a table is given below:

Type of Symmetry	Number of Unknowns	Memory usage	Solution Time in seconds
Symmetry not used	792	9.74 MByte	24.8
All 3 coordinate planes declared as planes of geometric symmetry	792	9.74 MByte	22.1
Plane $x = 0$ declared as electric plane of symmetry, plane $y = 0$ as magnetic plane of symmetry and plane $z = 0$ as geometric plane of symmetry	198	2.56 MByte	8.9

This example has relatively few unknowns. Most of the computational time is therefore used to determine the matrix elements in comparison to the time taken to solve the matrix equation. For applications with more unknowns, the reduction of unknowns could make a considerable difference in the time and memory required.

2.4.5 Special enforcement of symmetry: Even – odd method

The table in the previous section demonstrates the advantage of using electric and/or magnetic symmetry. For very large structures which have only geometrical symmetry, it may be worthwhile to consider two separate problems with electrical and magnetic symmetry as described below.

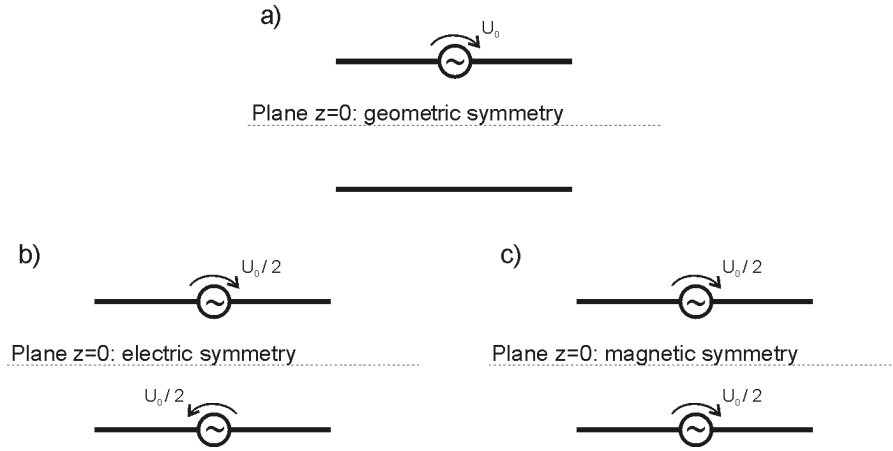


Figure 2-10: Separating a problem with geometrical symmetry into two problems with electric and magnetic symmetry respectively.

Figure 2-10 (a) shows the original problem. It consists of a dipole antenna with a passive wire below it. This is admittedly a very simple problem, normally this procedure would only be applied to much more complex structures.

The structure in figure 2-10 is symmetric about the plane $z = 0$, but the excitation is asymmetric and thus only geometric symmetry can be applied in FEKO. This problem may be separated into the two sub-problems shown in figures 2-10 (b) and (c). These two problems can be solved using respectively electric and magnetic symmetry about the plane $z = 0$. Each of these problems require only half the number of unknowns required for case (a). Superposition of the problems (b) and (c) — which must unfortunately be done manually as FEKO cannot do it automatically at the moment — yields the original problem.

The solution of each of the sub-problems requires only half of the storage space required for case (a). For very large problems the solution time is dominated by the time required to solve the system of linear equations. In this case the two sub-problems each requires only one eight of the time required for case (a).

2.5 Dielectric solids

There are six possible ways to model dielectrics, two of these apply to arbitrary bodies:

- In the surface current method the surface is subdivided into triangular surface elements. Basis functions are applied to these elements for the equivalent electric and equivalent magnetic surface currents. Boundary conditions result through the use of equivalent sources.

  The geometry is entered through the use of the ME card (section 8.2.24). By using the ME card it can be specified whether the triangular elements are to be used for metallic surfaces or dielectric surfaces. It is possible to use triangles and segments defined as metals within the dielectric.

  The ME card specifies the respective dielectric media on the two sides of the dielectric and uses the normal vector of the triangles to distinguish the two sides. Therefore the normal vectors should be checked with WinFEKO. `Example_04` (*Examples Guide*) shows a simple dielectric sphere. A more complex geometry, where a dielectric body is in contact with a conducting body, is described in `example_23`. Further details can be found there.

  The material properties are assigned through the use of the DI card (section 9.2.22).

- In the volume current method the volume is subdivided into cuboids through the use of a QU card (section 8.2.31). Each element can be assigned a different material property. Inside the element the polarisation current is assumed unknown.
  NOTE: If both $\varepsilon_r \neq \varepsilon_{r\ \text{environment}}$ and $\mu_r \neq \mu_{r\ \text{environment}}$ at any position, then two elements have to be assigned to that position, one for the dielectric property and one for the magnetic property.

  In `example_09` (*Examples Guide*) a dielectric cube is analysed with the volume current method.

The difference in the two methods is listed below:

Surface Current Method	Volume Current Method
the surface has to be discretised	the volume has to be discretised
homogeneous solids only	inhomogeneous solids possible

Apart from these two general formulations, there are a number of special solutions for dielectric bodies:

- Dielectric half-space e.g. ground surface:
  In this case the reflection coefficient method is used (see the BO card, section 9.2.17).

- Spheres consisting of one or more dielectric layers:
  A special Green's function is available (see the GF card, section 9.2.27).

- Planar multilayer substrate (with or without a perfect conducting ground plane):
  A special Green's function is available (see the GF card, section 9.2.27).

- Thin dielectric sheets:
  The volume equivalence principle is applied and the resulting equivalent currents approximated by a surface current (see the SK card, section 9.2.38).

The entered structures e.g. metallic wires and surfaces, do not necessarily have to be embedded in free space. The EG card (section 8.2.12) can be used to specify the material parameters of the surrounding medium.

2.6 Dynamic memory management

2.6.1 Setting maxalloc for out-of-core solutions

FEKO has the ability to manage the required memory dynamically, i.e. the memory required for the geometry data and matrix equations is determined and allocated at run time. When FEKO tries to allocate memory, it cannot determine the difference between RAM and virtual memory (system swap space). Due to the access times required for virtual memory and the random nature of memory access, the in-core solution is very inefficient when used in this way. FEKO also has an out-of-core solution which uses the data on disk in a much more efficient way. (The out-of-core technique is also used, of course, if the problem requires more memory than is available in both RAM and virtual memory.)

For solutions which do not fit into the available RAM, but do fit into the RAM plus virtual memory, the user should set `#maxallocm` to inform FEKO on the amount of available RAM (in MByte). Note that one should reserve some memory for the operating system and other running applications. For example, if a big problem must be executed on a PC with 512 MByte of RAM, `#maxallocm` should be set to about 460 MByte. This is done with the line

```
#maxallocm = 460
```

anywhere in the `*.pre` file (preferably near the start). For parallel versions of FEKO this memory limit applies to each process.

The following is a list of the variables which the user might occasionally set for special purposes such as the in-core limit described above.

`#maxallocm`	This sets the maximum allocatable memory in Mbytes. For example the definition `#maxallocm = 400` will allow a maximum of 400 MByte of memory to be allocated. If this is not enough, the matrix will be saved to the hard disk or the program will be halted. For parallel versions of FEKO this memory limit applies to each process. Note that if `#maxallocm` is used, it will have preference, and `#maxalloc` will be ignored if it is used in the same `*.pre` file.

`#maxalloc`	This sets the maximum allocatable memory in bytes. For example the definition `#maxalloc = 100*1024*1024` will allow a maximum of 100 MByte of memory to be allocated. This parameter has been replaced by `#maxallocm` — see above — but it is still understood for backwards compatibility purposes.

2.6.2 Other variables that are under user control

In some cases memory blocks have to be allocated for data storage before FEKO knows exactly how big these memory blocks have to be. In these cases it uses an estimate calculated by PREFEKO. If the estimated size is too small, FEKO will stop execution and give an error message. The appropriate size now has to be declared in the `*.pre` file. This is done by entering, for example, the line `#maxnv = 100` as for normal variables — see also section 6.3 — anywhere in the `*.pre` file.

`#maxnp`	The maximum number of columns and rows which a block in the matrix consists of in the Block-Gauss algorithm which solves the matrix equation. Dynamically `3*#maxnp*#maxnp*16` bytes are allocated for 3 blocks in the matrix.
`#maxnv`	The maximum number of connection points between wires and surface triangles.

2.6.3 Variables that are automatically set correctly

Note that normally PREFEKO estimates the following variables correctly and they should only be declared in cases where there is an explicit error message stating that larger memory blocks are required. Under normal circumstances these variables should not be set, they could have a negative impact on the FEKO performance.

`#maxaeedges`	The maximum number of edges between triangles that may be excited with the AE card.
`#maxanr`	The maximum number of sources.
`#maxapo`	The size of the memory block that is used to save the coefficients in the physical optics approximation. For `#maxapo=0` the necessary amount will be dynamically allocated.
`#maxarang`	The maximum number of ϑ or φ angles used with the AR card (excitation by a point source with a specified radiation pattern).
`#maxarpat`	The maximum number of radiation pattern excitations (AR card) allowed simultaneously.
`#maxbsobnr`	To accelerate the ray path search with PO the area under consideration is divided into boxes. Information pertaining to which box contains which object must be stored. A field of size `#maxbsobnr` is used in this case.

`#maxckant`	The maximum number of cut edges per cut in LFFEKO.
`#maxcolayer`	The maximum number of layers on a CO card using DOCOVR=3 or 4.
`#maxdels`	The maximum number of star basis functions that are deleted in LFFEKO. For each surface one star function has to be deleted.
`#maxdrnv`	The maximum number of triangle elements that can be connected to a segment at an attachment point.
`#maxfepkts`	The maximum number of points considered for the near field computation with the FE card when FELKOR=6 (irregular separately specified field points).
`#maxfoge`	The maximum number of areas that are described by using the Fock theory.
`#maxgfmsia`	The maximum number of entries in the interpolation tables for the Green's function of a planar substrate.
`#maxhacards`	The maximum number of HA cards (used internally to set up microstrip ports) that may be present in the `*.fek` file.
`#maxkanr`	The maximum number of "internal" edges (also the number of basis functions) per triangle. It may be larger than 3 if more than two plates share an edge.
`#maxknonr`	The maximum number of nodes that may lie against a segment.
`#maxl4cards`	The maximum number of L4 type loads.
`#maxlab`	The maximum number of labels.
`#maxlecards`	The maximum number of LE cards (which specify a load on an edge between triangles).
`#maxleedges`	The maximum number of edges between two surface triangles that can be loaded with a single LE card.
`#maxlengz`	Dimension of the interpolation table used for the planar multilayer Green's functions. This variable determines the maximum number of sample points in the z direction.
`#maxmedia`	The maximum number of different media used for the treatment of dielectric bodies in the surface equivalence principle. The surrounding free space (medium 0) is not counted (i.e. with `#maxmedia=1` one dielectric body can be treated).
`#maxndr`	The maximum number of triangles.
`#maxnka`	The maximum number of edges between two triangles.
`#maxnkapo`	The maximum number of edges in the physical optics approximation.
`#maxnkl`	In LFFEKO the maximum number of edges, that can be connected to a node, i.e. the maximum number of loop basis functions per edge.
`#maxnkno`	The maximum number of nodes between segments.
`#maxnlayer`	The maximum number of layers for the special Green's function of a planar substrate.

`#maxnlp`	The maximum number of loops, that may be formed in LFFEKO.
`#maxnqua`	The maximum number of dielectric cuboids.
`#maxnseg`	The maximum number of segments.
`#maxntetra`	The maximum number of tetrahedral volume elements for a FEM solution.
`#maxnzeile`	The maximum number of basis functions in the moment method area.
`#maxpoka`	The maximum number of bordering edges to the PO area.
`#maxpokl`	The maximum number of wedges in the PO area.
`#maxpolyf`	The maximum number of polygonal surfaces that can be used to represent a body in the UTD region.
`#maxpolyp`	The maximum number of corner points allowed for a polygonal plate.
`#maxpovs`	The maximum number of label to label visibility specifications set by VS cards (a card with a range sets a number of entries equal to the size of the range).
`#maxsklayer`	The maximum number of layers at an SK card.
`#maxtlcards`	The maximum number of TL cards.
`#maxutdzyl`	The maximum number of cylinders in the UTD region.
`#nmat`	The memory size that may be allocated for the matrix of the system of linear equations. For `#nmat=0`, the necessary amount will be allocated dynamically. The allocation is not specified in bytes, but in terms of the number of type DOUBLE COMPLEX numbers. (These require 16 bytes each.) For example, 400 MByte is specified by setting `#nmat = 400*1024*1024/16`. The same effect can be achieved by setting the variable `#maxalloc` such that it is unusual to assign a value to `#nmat`.
`#npuf`	The maximum number of control cards that may occur in a loop (for example in a frequency sweep).

2.7 Environment variables

This section lists the environment variables that may be used to control the execution of FEKO. See also the discussion of the installation (the script `initfeko` or the batch file `initfeko.bat`) in the *Getting Started* manual. This script and batch file is usually automatically created by the FEKO installation program, and the environment variables are set correctly. Therefore the user does not need to set environment variables manually.

The following environment variables may be set.

`FEKO`	Must be set to the path where the FEKO executables are located, which normally is `FEKO_HOME/bin`.

FEKO_CMDINFO	If this environment variable is set to the value 1, FEKO writes additional data concerning the number and the value of the received command line parameters to the screen. This can be useful to trace errors in the parallel version of FEKO used in connection with some implementations of `mpirun`, `mpiopt`, `mpprun` etc.
FEKO_LITE	If this environment variable is set to 1, FEKO runs in a restricted LITE mode (see the *Getting Started* manual).
FEKO_HOME	This variable is set to the FEKO installation path which contains the subdirectories such as `bin`, `doc`, `license` and, for the parallel version, `mpi`.
FEKO_MACHFILE	The parallel version of FEKO is started by running RUNFEKO with options `-np x`. When FEKO is installed on a parallel computer or a computer cluster, the configuration of the cluster and the number of processes that should be run on each computer is specified during the installation. This can can be overwritten for any FEKO run by creating a so-called *machines file* and setting the environment variable `FEKO_MACHFILE` to point to this file. (More detail can be found in section 7.2.)
FEKO_MACHINFO	If this parameter is set, FEKO will write information about the machine precision to both the screen and the output file.
FEKO_MPISTATISTICS	This environment variable provides additional information about the performance of the parallel version of FEKO. There are three options 1 Give a detailed report of the CPU and run times for the individual processes. It is, for example, possible to determine how much time each process required during the computation of the array elements. 2 Give as additional output the MFLOPS rate of each process (without network communication time). This is useful to determine the relative performance of nodes in a heterogeneous cluster. 4 Give information about the network performance (latency and bandwidth). This is very useful when configuring parallel clusters. The options can be added in a binary fashion, for example setting `FEKO_MPISTATISTICS=5` will print both the run times and network performance.

FEKO_SYSINFO

If this parameter is set, FEKO provides information about the computer system (mostly used on UNIX platforms).

FEKO_TMPDIR

This variable specifies the directory where FEKO will write paging files, when using the out-of-core solution. In the past it was required that the definition ended in a backslash (Windows) or a slash (UNIX). This is no longer required. For example, in UNIX it may be set with

```
set FEKO_TMPDIR=/tmp
export FEKO_TMPDIR
```

FEKO_USER_HOME

This directory is used to write user specific initialisation files. This variable replaced FEKO_WRITE. It is provided to allow different users to save unique configurations, or for when the user does not have write access to the FEKO directory. For Windows systems this is normally the same as FEKO and on UNIX systems it is usually $HOME\.feko.

FEKO_WHICH_MPI

FEKO uses different MPI implementations for the different platforms and thus the different platforms require different command syntax to start FEKO. RUNFEKO provides an interface that remains the same on all platforms. However, it must know which MPI implementation is used. This is done by setting the environment variable FEKO_WHICH_MPI (it is automatically set by the installation script) to one of the following options:

0 Not initialised
1 mpich (for example, Linux, SUN, IBM)
2 HP-MPI
3 NEC-MPI on NEC SX/5
4 SGI-MPI
5 CRAY/SGI-MPI on CRAY T3E
6 ScaMPI on Linux ix86 and Alpha
7 Compaq MPI on Tru64 UNIX
8 SCore on Linux Myrinet clusters
9 GM on Linux Myrinet clusters

FEKO_WRITE

This variable is obsolete and is replaced by FEKO_USER_HOME which has the same functionality on both Windows and UNIX systems.

FEKO_WRITE_RHS	If this environment variable is set (value arbitrary), FEKO writes the right side of the set of linear equations to a *.rhs file. This is only useful for test purposes, such as when one wants to analyse this vector with another program.
FEKOLANG	Selects the language of operation. This must be either d for German, or e for English.
FEMAP	The directory where the CAD/mesher FEMAP is installed.
OMP_NUM_THREADS	This is set to 1 and is required by the libraries unsed in the FEKO kernel.

2.8 Checking the validity of the results

If a calculation has been done with FEKO, the results have to be checked. There are a number of ways of doing this:

- a comparison with exact results, if these are available
- a comparison with results that have been published in the literature
- a comparison with another program that is based on another method of calculation.
- a comparison with measured results
- plausibility, e.g. negative real input impedances do not exist.

If these possibilities are not available, then the following should be tried:

- After a normal calculation with FEKO run another pass with a finer mesh (see IP card, section 8.2.17). The number of segments should be at least 1.5 times greater than with the normal calculation. If there is a large difference in the results, then an error has occurred.
- Doing a power balance. The power fed into an antenna through the power source must be equal to the radiated power. The radiated power can be calculated by integrating the power flux density. This is done by using the FF command (see example_07 or example_08 in the *Examples Guide*).

3 The program WinFEKO

The program WinFEKO has been developed to serve as a Graphical User Interface (GUI) for FEKO in a MS Windows environment.

WinFEKO uses the dynamic link libraries `OpenGL32.dll` and `Glu32.dll` which should be available on all NT/2000/XP systems and most 95/98/ME systems. (The libraries for Windows 95/98/ME are included on the FEKO CD in the directory `utils\OpenGL95`. If they are not available on your system, copy them from the CD to your FEKO installation directory.) WinFEKO also use the libraries `Glut32.dll` and `GLPrint.ocx` which are installed with FEKO. In addition it calls the program GraphFEKO developed for the extraction of data from the FEKO output file (see section 5).

A basic description of WinFEKO is presented in this chapter. See the *Getting Started* manual to get an introduction to using FEKO and all the utility modules, including WinFEKO.

3.1 Hardware and software requirements for WinFEKO

3.1.1 Operating systems

WinFEKO has been developed for the MS Windows 95/98/ME and NT/2000/XP environments. It has been our experience that the OpenGL libraries used for visualisation in WinFEKO were more stable under Windows NT/2000/XP than under 95/98/ME.

On clusters of PCs running MS Windows NT/2000/XP, the parallel version of FEKO is started from WinFEKO (see section 3.5.4). This uses MPICH.NT which requires Windows NT/2000/XP for all the machines in the cluster and the PC where the job is started from, even if the this PC does not form part of the parallel cluster.

3.1.2 Memory requirements

It is recommended that at least 64 MByte RAM are available on the system (more memory would be required for large models and data sets). Ensure that a large enough swap file exists on systems with 64 MByte RAM or less.

3.1.3 Display settings

WinFEKO can be used with 256 colours, but a colour setting of at least 16 bits is recommended for effective visualisation. The development of WinFEKO was done for a 800×600 screen resolution. A higher resolution may be used, but a lower resolution is likely to give trouble.

3.1.4 Graphics card

With properly installed graphics cards that support *3D* OpenGL hardware rendering, the user should see a considerable speedup in visualisation (see graphics card documentation for information on OpenGL support). Note that some cards only support OpenGL at a certain colour depth. A number of graphics cards with 3D hardware rendering do not support all OpenGL features. Users might thus experience problems with some of the WinFEKO features. Setting the system colour depth to 256 colours forces software rendering with which OpenGL should not give any problems. Alternatively one may rename the file `OpenGL32_soft.dll` in the `bin` subdirectory to `OpenGL32.dll` to force software rendering at any colour depth.

3.2 Running WinFEKO

To run WinFEKO, select *Programs* → *FEKO* → *WinFEKO* from the Windows *Start* menu if the FEKO shortcut has been installed in the default folder. On startup WinFEKO tries to read a file `winfeko.ini` from the user's home directory specified by the environment variable `FEKO_USER_HOME`. This file contains information regarding the previous execution of WinFEKO (e.g. which file was loaded the previous time WinFEKO was executed).

WinFEKO can be executed with a `*.pre` or `*.wfp` (**win**feko **p**roject) file as command line parameter. For example:

```
winfeko example_01.wfp
```

or just

```
winfeko example_01
```

In such a case WinFEKO will ignore the previous file information in the `winfeko.ini` file and load the project `example_01.wfp`.

3.3 Hot-keys in WinFEKO

WinFEKO uses certain key combinations (or hot-keys) to allow faster access to functions that are used often. Some of these keys are always active, but most are only active when the *Main display options* panel is open (i.e. when no results or selection panel is shown).

The following keyboard keys are available as hot-keys in WinFEKO.

Always active

<F2>	Save Project
<F3>	Open Project
‘	Return to *Main display options* panel
<Alt><1>	Run FEMAP
<Alt><3>	Display neutral file

Only active with Main display options panel visible

<1>	Run EditFEKO
<2>	Run PREFEKO
<3>	Display FEK model
<4>	Run FEKO solver
<5>	Load and display output file
<6>	Start GraphFEKO
<U>	Rotate model up
<B>	Rotate model down
<L>	Rotate model left
<R>	Rotate model right
<A>	Toggle axes
<E>	Toggle DP label display
<N>	Toggle element number display
<S>	Toggle label number display
<O>	Toggle element normal display
<V>	Toggle hidden line removal
< + >	Zoom in (on the numerical keypad only)
< − >	Zoom out (on the numerical keypad only)
Arrow R	Pan right
Arrow L	Pan left
Arrow U	Pan up
Arrow D	Pan down

3.4 Toolbars in WinFEKO

The toolbars in WinFEKO can be organised into 5 main groups. Most of the speed buttons on these toolbars are associated with menu items and their functionality will be described in the corresponding menu item section. The functionality of speed buttons not associated with menu items will be discussed in more detail in this section.

3.4.1 FILE control toolbar

Used for WinFEKO file control and printing.



1. Create a new project (see section 3.5.1.1).
2. Open project (see section 3.5.1.2).
3. Save project settings (see section 3.5.1.4).
4. Print the model (see section 3.5.1.6).

3.4.2 FEKO control toolbar

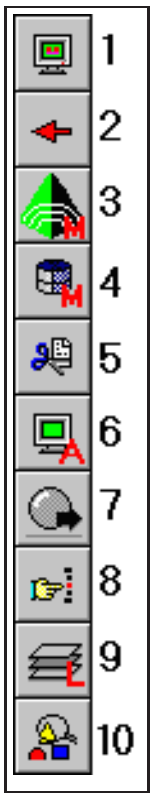
Used for execution of FEMAP and the different FEKO modules.



1. Run FEMAP (section 3.5.2.1).
2. Run EditFEKO (section 3.5.2.3).
3. Run PREFEKO (section 3.5.2.4).
4. Display model (section 3.5.2.5).
5. Display field points (section 3.5.6.7).
6. Display excitation (section 3.5.6.8).
7. Run FEKO solver (section 3.5.3.1).

3.4.3 DISPLAY OPTIONS control toolbar

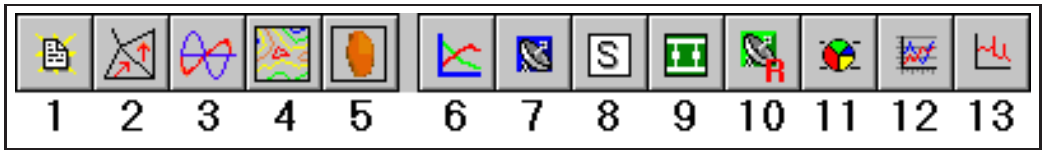
Used to quickly go to the different model display option panels available in WinFEKO.



1. Go to *Main display options* panel (see section 3.5.6.1).
2. Go back to previous options panel (see section 3.5.6.2).
3. Go to *FEK file display options* panel (see section 3.5.6.3).
4. Go to *NEU file display options* panel (see section 3.5.6.4).
5. Go to *Cutplane options* panel (see section 3.5.6.5).
6. Go to *Advanced visibility options* panel (see section 3.5.6.6).
7. Go to *Transformation control* panel (see section 3.5.7.2).
8. Go to *Find element* panel (see section 3.5.7.1).
9. Toggle label display. This button sets the *Colours* option in the *FEK file display options* panel to *Label number* and activates the *Legend* under *Visibility* on the *Main display options* panel. This provides a quick way to display label numbers. If the *Colours* option in the *FEK file display options* panel is already selected when this button is clicked, label display is switched off.
10. Go to *Entity selection* panel (see section 3.5.7.4).

3.4.4 RESULTS control toolbar

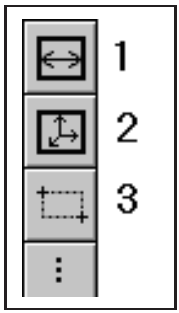
Used to load and display data obtained from the output file.



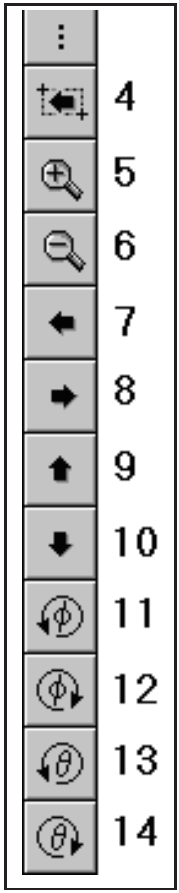
1. View output file (see section 3.5.5.1).
2. View currents (see section 3.5.5.12).
3. View 3D polar patterns (see section 3.5.5.7).
4. View near field ortho-slices (see section 3.5.5.10).
5. View near field iso-surfaces (see section 3.5.5.9).
6. Start GraphFEKO (see section 5).
7. Antenna Parameters (see section 3.5.5.2).
8. S-Parameters (see section 3.5.5.3).
9. Network Parameters (see section 3.5.5.4).
10. Receiving Antenna Parameters (see section 3.5.5.5).
11. 2D Far field plots (see section 3.5.5.6).
12. 2D Near field plots (see section 3.5.5.8).
13. Open GraphFEKO to extract continuous frequency results.

3.4.5 RENDER control toolbar

Used to control the model rendering.



1. Normalise to extent. With this button, the zoom factor is set such that the complete model will fit into the display window. The extent is calculated from all geometrical elements in the model, including the node points (DP card) and the axes, as well as the requested field points in the model.
2. Isometric view and normalisation. The same as *Normalise to extent*, but the view direction is set to the default ($\vartheta = 65°$, $\varphi = 35°$).
3. Zoom to window. Click this button to zoom to a specific rectangular window. After clicking the button do the following: left-click, hold down and drag on the display to select the window to zoom to. Release the left-mouse button to execute the zoom to window.



4. Previous view. Returns the display to the previous view. Zoom factor, view angles and pan positions are "remembered".
5. Zoom in by a constant factor. The default constant factor is 25% but it can be set under **Transformation control — Zoom** (see section 3.5.7.2).
6. Zoom out by a constant factor. The default constant factor is 25% but it can be set under **Transformation control — Zoom** (see section 3.5.7.2).
7. Pan left by a constant factor. (The default constant factor is 10; 100 represents panning horizontally across the display screen).
8. Pan right by a constant factor. (The default constant factor is 10; 100 represents panning horizontally across the display screen).
9. Pan up by a constant factor. (The default constant factor is 10; 100 represents panning vertically across the display screen).
10. Pan down by a constant factor. (The default constant factor is 10; 100 represents panning vertically across the display screen).
11. Increase Phi view angle by a constant factor (15 degrees).
12. Decrease Phi view angle by a constant factor (15 degrees).
13. Increase Theta view angle by a constant factor (15 degrees).
14. Decrease Theta view angle by a constant factor (15 degrees).

3.5 Main menu structure

3.5.1 File menu

This is used for project (or file) and printing control.

3.5.1.1 New project

Select this item to create a new FEKO project. On selection the **Create new project** panel is activated.

Decide on a name for the new project (for example, `dipole`) and enter it into the *Project name* field. The name *can* be longer than 8 characters. Type the directory path in the *Project directory path* field (or Browse to select a directory path).

Decide on the type of problem. Three different "templates" are available at present:

- The *Radiation* template will create a FEKO input file (`*.pre` file) with the control cards typically used with radiation problems.
- The *Scattering* template will create a FEKO input file (`*.pre` file) with the control cards typically used with scattering problems.
- The *None* template will create a FEKO input file without any control cards.

Click the *Create project* button. A new directory will be created in the selected path. The name of the directory will be the selected *Project name* (e.g. `dipole`). In this directory three files will be created with the same name as the directory (or project name). The three files are the FEKO input file (`*.pre` file), a WinFEKO project file (`*.wfp`) and a FEMAP model file (`*.mod`).

3.5.1.2 Open project

Select the *Open project* item to open an existing FEKO project. The project that is open at the time (if any) will first be closed. The default extension when browsing for the project to open, is `*.wfp` (WinFEKO project extension).

Select a different file type if no `*.wfp` file exist. For example, if only PREFEKO input files (`*.pre` files) exist, select *PREFEKO file* as the file type.

If a `*.wfp` file is selected, information regarding this project is loaded (for example, the previous zoom factor, view angles, etc.). If a `*.pre` file or `*.fek` file is loaded and no `*.wfp` file exists, the default program settings are chosen for the model. In such a case, it is recommended that the project be saved immediately — select *File* → *Save project*. This will create a `*.wfp` file to associate with this `*.pre` (or `*.fek`) file.

WinFEKO searches for a `*.fek` file (in the current directory) with the same name as the selected `*.wfp`, `*.pre` or `*.fek` file. If the `*.fek` file is found, the model information is loaded from the `*.fek` file and displayed. If not, the message `No FEK file available for this project` is displayed at the bottom of the display screen. Run PREFEKO to create the `*.fek` file.

3.5.1.3 Close project

Select *Close project* to close the current project. The project information is saved in the `*.wfp` file. WinFEKO releases all memory allocated when the project was opened, the `*.fek` file was loaded and/or the output file information was loaded.

3.5.1.4 Save project

Select *Save project* to save settings associated with the current project. The settings, such as zoom factor, view angle, pan position, axis length, cut-planes and all the display options on the **Main display options** panel are saved in the `*.wfp` file.

3.5.1.5 Save project as

Select *Save project as* to save the project under a new name. This is very different than *Save project*. Here a new directory is created for a completely new project.

Decide on the name the project should be saved as. Type it into the *Project name* edit box. A new directory will be created in the selected path (use the *Browse* button to select the path if necessary). The name of the new directory will be the selected *Project name*. For example, if the current project name is `dipole`, and the new (or "save as") project name is selected as `dip_2`, then a new directory `dip_2` will be created.

Decide on the *Copy options*. Three different options are available, at present:

- *All files:* All the files in the current project directory (e.g. `dipole`) will be copied to the new directory (e.g. `dip_2`). Also, all files in the current project directory with the name `dipole.*` will be renamed as `dip_2.*` in the new directory (e.g. `dipole.pre`, `dipole.fek`, etc. are renamed as `dip_2.pre`, `dip_2.fek`, etc.
- *Model files:* Only the "model" files in the current project directory (e.g. `dipole.pre`, `dipole.mod`, `dipole.neu`, `dipole.fek`) will be copied to the new directory **and** renamed.
- *Pre files only:* Only the `*.pre` file in the current project directory (e.g. `dipole.pre`) will be copied to the new directory **and** renamed.

Click the *OK* button. This will start the copying and renaming. The current project is closed and the new project is loaded.

3.5.1.6 Printing

Select *Print* to print the model display. On selection the **Winfeko Printing** window is activated. Select printing to *Printer*, *File* or *Clipboard*. A standard windows printer setup may be activated by selecting *Printer setup*.

A *Quality* option can be selected. *Presentation* quality results in a better quality print (especially when printing to a colour printer), but more memory (approximately 16 times more than *Fast*) is required. It also takes longer to print *Presentation* quality. *Normal* quality is a good compromise, requiring four times more memory than *Fast* and being faster than *Presentation* quality. The paper size (Default or A4) can be selected.

On printing to a printer, file or clipboard the following options are also available:

- Grayscale: A completely new model with Grayscale colouring is printed.
- Reverse: The background is white (instead of black) and all white pixels or lines are drawn in black.
- Project Info: Information relating to the current project will also be printed. When printing to a printer, the *Frame* option is automatically enforced if this option is selected. Not available for clipboard printing.
- Frame: Draws a frame around the model. Not available for clipboard printing.

On printing to the clipboard, select either the *Bitmap* or *Metafile* option. *Metafile* clipboard printing is vector based and the image created would not necessary appear exactly as on the rendering screen in WinFEKO.

Click the *Preview* button to get a preview and to enable the *Print* button. The model, as it will be printed to a file or printer, is displayed. All rendering of the model is now disabled. To change the model size, position (or any other setting) click the *Cancel* button and manipulate the image. Select *Print* and *Preview* again to return to the "previewing" of the model.

Click the *Print* button to send the selected model display to a printer, file or clipboard.

When printing to a file a *Print To File* dialogue box is opened. Select a file name and directory path. Click the *Save as type* drop down list to selected one of the following "bitmap based" graphics file types:

- Bitmap: Prints to a bitmap file with extension `*.bmp`. The bitmap file created could be quite large (in memory) for Presentation mode printing.
- Jpeg: Prints to a jpeg file with extension `*.jpg`. A jpeg file will be much smaller in memory than a bitmap file, but picture quality will be lost.
- Gif: Prints to a gif file with extension `*.gif`. A gif file is a good choice. Much smaller in memory than a bitmap file, but better quality than jpeg.
- Eps: Prints the bitmap to an image embedded in an encapsulated postscript file `*.eps`. These eps files can become very large as it is a bit by bit representation of what is on the rendering screen, encoded into the postscript language.
- Tif: Prints to a tif file with extension `*.tif`. A tif file is of good qualtity but could, similar to a bitmap, be quite large.

Also available from the *Save as type* drop down list is one of the following "vector based" graphics file types. The image created with vector based printing would **not** necessary appear exactly as on the rendering screen in WinFEKO.

- Vector Postscript: This creates a vector based postscript file `*.ps` from all the objects in the WinFEKO model that represents (as close as possible) the image rendered on the screen.
- Enhanced Metafile: Same as vector based postscript printing but printing to an Enhanced Metafile `*.emf`.
- Metafile: Same as vector based postscript printing but printing to a Windows Metafile `*.wmf`.

3.5.1.7 List of previous projects

Below the *Printer setup* item, is a list of previous projects. Select one of these to quickly open one of the projects recently worked on.

3.5.1.8 Exit

Select this item to exit WinFEKO. On exit, the `winfeko.ini` file is updated for use when WinFEKO is executed the next time.

3.5.2 Preprocessing menu

Used for the various preprocessing functions and module executions required before the FEKO solver can be started.

3.5.2.1 Run FEMAP

Select this item to start the CAD/mesher FEMAP. FEMAP is started with the `*.mod` file associated with the current project. For example, if the current project name is `dipole`, FEMAP is started with the file `dipole.mod` if such a file exists in the current project directory path. If not, FEMAP is started with the default FEMAP startup settings (untitled file).

The FEMAP environment variable must be set to the correct directory path for FEMAP to start successfully from the WinFEKO environment. This should have been done automatically by the WinFEKO installation program. (See the *Getting Started* manual.)

The user can also execute another 3rd party CAD packages of his choice using this menu item. The path and extension of such a package must be specified by the user under the General Settings options (see section 3.5.8.1).

3.5.2.2 Display FEMAP *.neu

If a neutral file (`*.neu`) has been created (exported) from FEMAP and the neutral file name is the same as the current project name (e.g. `dipole.neu`), then the meshed geometry in this neutral file is loaded and displayed. At the bottom of the display window an identification legend (**NEU File**) is drawn. (At present WinFEKO only displays line and triangle elements present in the neutral file.)

3.5.2.3 Run EditFEKO

Select the *Run EditFEKO* item to start the program EditFEKO. EditFEKO is started with the `*.pre` file associated with the current project (e.g. `dipole.pre`). If no such file exists EditFEKO will still be started but with an empty file.

The FEKO environment variables must be set correctly for EditFEKO to start successfully from the WinFEKO environment. This should have been done automatically by the FEKO installation program. (See the *Getting Started* manual.)

3.5.2.4 Run PREFEKO

Select the *Run PREFEKO* item to run PREFEKO on the `*.pre` file associated with the current project (similar to typing `prefeko.exe dipole.pre` at a command prompt).

The program PREFEKO is executed in a minimised command prompt. PREFEKO program feedback is passed to a `*.lg1` file (e.g. dipole.lg1) which is displayed in the **PREFEKO log file** window, opened when the *Run PREFEKO* item was selected. Click on the **Close** button (top-right of **PREFEKO log file** window) to close this window.

The `FEKO` environment variable must be set correctly for PREFEKO to start successfully from the WinFEKO environment. This should have been done automatically by the FEKO installation program. (See the *Getting Started* manual.)

3.5.2.5 Display FEKO model

The `*.fek` file associated with the current project is loaded and displayed, when this item is selected. A FEKO model file (e.g. `dipole.fek`) is created as output when the program PREFEKO is executed. If no `*.fek` file (with the project name) is available in the current project path, a standard **open** dialog is activated. This allows for the selection of any other `*.fek` file.

If WinFEKO experiences problems on trying to load the selected `*.fek` file, no FEKO model will be displayed. An error or warning should appear and this will give a hint to what might be wrong with the `*.fek` file.

On successful loading and display of a `*.fek` file, an identification legend (**FEK File**) is drawn at the bottom of the display window.

3.5.2.6 Clear

This item "clears" all information associated with the geometry and output data of the current project. The project is **not** closed, but no FEK or NEU model will be displayed. Effectively this command releases all memory allocated when loading the FEK or NEU file geometries, or the output file data.

It is recommended to use this command before running the FEKO solver (from within WinFEKO) on large FEKO problems.

3.5.3 Solve menu

3.5.3.1 Run solver

Select the *Run solver* item to execute the sequential FEKO solver on the `*.fek` file associated with the current project. The launcher `runfeko.exe` is executed in a command prompt. If an output file with the current project name (e.g. `dipole.out`) already exists in the current directory path, a warning appears before the FEKO solver is executed. Select *Yes* to overwrite the existing output file and *No* to cancel the *Run solver* command.

The FEKO environment variables must be set correctly for FEKO to start successfully from the WinFEKO environment. This should have been done automatically by the FEKO installation program. (See the *Getting Started* manual.)

3.5.4 Running the parallel version of FEKO

On Windows NT/2000/XP systems the parallel version of FEKO is also launched from the *Solve* menu in WinFEKO. It uses MPICH.NT for the communication between the different hosts/nodes. Both FEKO and MPICH.NT must be installed on all the PCs in the cluster and the executable `feko.csv.exe` must be located in the same directory on each node. (If FEKO is installed with parallel support, MPICH.NT is installed automatically.) The parallel version of FEKO must be activated on all the nodes in the cluster, and WinFEKO must be activated on the local PC.

The current user on the PC where the job is launched must have an account on all the machines where he intends to start the parallel job. All these accounts must use the same password.

3.5.4.1 Configuring MPICH

Before running a parallel job, MPICH.NT must be configured properly. A default machines file (see section 7.2.2) was created during installation. This may also be configured for the local user. Select *Solver* → *Parallel version* → *Configure* from the main menu to open the configuration window. The configuration window allows the user to specify the host names and the number of processes to run on each host. FEKO must be installed in the same location on each host. Usually one would run one process per CPU, which determines the number of processes for each host, for example 2 processes for a dual-board machine. One may also use this for a crude load balancing, running more processes on hosts with faster CPUs or more RAM. The *Add hosts line* button creates room to specify additional hosts.

We recommend that the parallel job is started from a PC that forms part of the cluster and that this host is listed first.[2]

If the "Check node performance" and "Check network performance" options are selected FEKO will also print a table giving the performance of the various nodes. It is recommended that this is used during setup to ensure an optimum configuration. (These checks are repeated each time FEKO calculates the solution, so it may require a significant amount of time if the test file contains multiple frequencies. One would not keep these options selected after the initial setup, except for debugging purposes.)

[2]It is possible to launch the job without including the local machine. The `*.fek` input file must then be located on the first PC in the list and the `*.out` file is created on this PC — both in the exactly same directory as the project directory on the local machine. It is *the user's responsibility to transfer the files between the local machine and the first machine in the list if these are not the same* — WinFEKO does not do this.

When the user clicks *OK* on this window, the hosts are saved to a file `machines.feko` in the directory specified by the environment variable `FEKO_USER_HOME`. The format of this file is the same as that of the default machines file created during installation. It is not required to run the configuration, but one may do so at any time to change the host list.

3.5.4.2 Execute

Once a proper `machines.feko` file has been created, FEKO is executed by selecting *Solver → Parallel version → Execute.* WinFEKO searches for a machines file in the order

```
%FEKO_USER_HOME%\machines.feko
%FEKO_USER_HOME%\machines
%FEKO_HOME%\mpi\share\machines.feko
%FEKO_HOME%\mpi\share\machines
```

and uses the first file that exists. It then calls RUNFEKO with the `-np x` option to start the parallel version. In this case `x` is calculated as the sum of the processes listed for the hosts in the machines file.

3.5.5 Results menu

This menu is used to load and display data obtained from the output file.

3.5.5.1 View output file

Select this item to load the output file. A **FEKO output file** read-only editor is opened and the **Output file display options** panel is activated.

Type in a string in the **Search for text** edit box. Hit the *Search* button to search for the text. The other buttons are mostly speed buttons for quick searching of a specific text string in the output file, for example when the *WARNING* button is clicked, the **Search for text** edit box is filled with the text string "WARNING" after which the *Search* button is automatically "clicked".

3.5.5.2 Antenna parameters

When this item is selected, the program GraphFEKO is executed with the current project `*.out` file passed as the first command line parameter. GraphFEKO starts and automatically selects the current project `*.out` file and loads the antenna parameter data (if it exists). The **Antenna parameters** panel in GraphFEKO is also activated automatically.

Select this item to load and display V_{in}, I_{in}, Z_{in}, Y_{in}, S_{11} and other antenna parameter data. For further information see section 5.

Note that every time this item is selected from within WinFEKO, the program GraphFEKO will be executed. If GraphFEKO is already running with the current project `*.out` file selected, rather use the *Import → Antenna parameters* menu item in GraphFEKO to load the antenna parameter data.

3.5.5.3 S-parameters

When this item is selected GraphFEKO starts, selects the current project `*.out` file and loads the S-parameter data as calculated on request of an SP card (see section 9.2.39). The **S-parameters** panel in GraphFEKO is also activated automatically.

If GraphFEKO is already running with the correct `*.out` file selected, rather use the *Import → S-parameters* menu item in GraphFEKO to load the S parameters.

3.5.5.4 Network parameters

When this item is selected GraphFEKO starts, automatically selects the current project `*.out` file and loads the network parameter data (if it exists). The **Network parameters** panel in GraphFEKO is also activated automatically.

Select this item to load and display S_{11}, S_{21} and the calculated currents in selected segments. For further information see section 5.

If GraphFEKO is already running with the correct `*.out` file selected, rather use the *Import → Network parameters* menu item in GraphFEKO to load the network parameter data.

3.5.5.5 Receiving antenna (Rx)

When this item is selected GraphFEKO starts, automatically selects the current project `*.out` file and loads the receiving antenna data (if it exists). The **Antenna reception** panel in GraphFEKO is also activated automatically.

Select this item to load and display the calculated currents in selected segments as a function of incident wave direction (or frequency). This data can only be extracted when an incident wave has been used as excitation. For further information see section 5.

If GraphFEKO is already running with the correct `*.out` file selected, rather use the *Import → Receiving antenna (Rx)* menu item in GraphFEKO to load the receiving antenna data.

3.5.5.6 Far fields → 2D plot

On selection of this item, GraphFEKO starts, automatically selects the current project `*.out` file and loads the far field data (if it exists). The **Far fields** panel in GraphFEKO is also activated automatically.

Select this item to load and display far field data such as gain, directivity, RCS, etc. For further information see section 5.

Note that every time this item is selected from within WinFEKO, the program GraphFEKO will be executed. If GraphFEKO is already running with the current project `*.out` file selected, rather use the *Import → Far fields* menu item in GraphFEKO to load the far field data.

3.5.5.7 Far fields → 3D polar plot

On selection of this item the far field data is loaded from the current project output file and the **3D radiation patterns** panel in WinFEKO is activated. The far field data is displayed in 3D, together with the model geometry. Right-click on the pattern to display information associated with the pattern (at the point of selection) on the rendering scene.

> **Note:** If any options are changed on the **3D radiation patterns** panel either the *Apply* or *OK* button must be clicked for the changes to take effect. The *Apply* button "registers" all changes and the 3D pattern plots are updated and re-displayed. The *OK* button does exactly the same as the *Apply* button, but also closes the **3D radiation patterns** panel and activates the **Main display options** panel. The *Cancel* button closes the **3D radiation patterns** panel and activates the **Main display options** panel, **without** "applying" any changes.

Select a frequency and far field block from the lists in the **Data block selection** group.

Select the *Line*, *Surface*, *Colour* or/and *Smooth* options under **Display options**. The *Smooth* option will only have an effect when the *Surface* option has also been selected. Unselect the *Surface* option if the pattern data in the current block has been calculated at one theta or phi position only.

The available **Far field quantity** and **Data component** can also be selected. To display the polarisation select *Axial R* or *L / R* under **Polariz. options**. *Axial R* plots the axial ratio in shaded colours. (Dark blue represents -1 and red 1.) *L / R* plots Left, Right or Linear polarisation direction. Blue represents left, Red represents Right, and Green represents Linear.

Select *Linear* or *dB* scaling under the **Scaling** options. *Min* (minimum) and *Max* (maximum) slide bars and edit boxes are available to clip the limits of the data. With *Scale blob* selected, the 3D pattern itself will change when changing the *Min* and *Max* settings. With *Scale blob* unselected, only the colour coding on the 3D pattern will change.

Adjust the 3D pattern *size* using the slide bar under **Blob settings**. Select the *Close blob* option to close a 3D pattern in the phi direction (e.g. if far field data has been calculated for phi=0 to 355, the 3D pattern will start at phi=0 and stop at phi=360 if the *Close blob* option has been selected).

Also under **Blob settings** are three edit boxes which can be used to move the *Origin* of the 3D pattern to an arbitrary point. Enter the x, y and z coordinates (in metres) and hit the *Apply* button for the origin shifting to take effect. With a vertex or geometric entity selected (see 3.5.7.4) the *Select Entity* button becomes enabled. Click this button to fill the x, y and z coordinate point edit boxes with the coordinates of the selected entity.

Unselect the *Elt. direction* option in the **Main display options** panel if a 3D pattern colour display is obviously asymmetrical.

The 3D patterns and model geometry can be made visible (invisible) by selecting (unselecting) the *3D pattern* and *Geometry* options in the **Main display options** panel.

3.5.5.8 Near fields → 2D plot

On selection of this item, GraphFEKO starts, automatically selects the current project `*.out` file and loads the *near field* data (if it exists). The **Near fields** panel in GraphFEKO is also activated automatically.

Select this item to load and display near field data such as electric field (E), magnetic field (H), power density (S), Specific Absorption Rate (in a lossy dielectric) etc. For further information see section 5.

Note that every time this item is selected from within WinFEKO, the program GraphFEKO will be executed. If GraphFEKO is already running with the current project `*.out` file selected, rather use the *Import → Near fields* menu item in GraphFEKO to load the near field data.

3.5.5.9 Near fields → Iso-surface

On selection of this item the near field data is loaded from the current project output file and the **Near field options** panel in WinFEKO is activated.

Note: See the comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

Note further: If the checkbox next to the **Near field options** caption is checked then the panel is "activated". This means that any changes made to the options available on the panel will be applied immediately. This is sometimes useful but with the panel "activated" rendering can become very slow.

Select a frequency and near field block from the lists under **Data block selection**. If the selected data block does not contain a 3D data set (i.e. near fields calculated for multiple x, y and z components) then a message **No 3D data in selected block** is displayed under the **Iso-surface setting**. No iso-surface will thus be calculated or displayed.

Lines, *Surface*, *Colour* and *Arrows* under **Display options** do NOT have any affect on the iso-surface display.

The *Iso Value* under **Iso-surface settings** can be set to select the value of the iso-surface. An iso-surface represents a surface of equal magnitude in a 3D data set. Iso-surface calculations for large sets of 3D data needs intense processing and could take some time even on relatively fast systems. A hint is to select the required quantity and component. Wait until the extent and default iso-surface value have been calculated (slide bar and edit box under **Iso-surface settings** will be updated). Then select a relatively high iso-value before hitting the **Apply** button. The iso-surface calculation should be relatively fast (except in the case of the phase component). Now set the iso-surface value smaller to the required value.

The available **Near field quantity** can also be selected. The Pointing vector (S) is only available when both E and H-fields have been calculated at the same positions (with one FE card). SAR is only available if the near electric fields in dielectric regions have been calculated. (See end of this section for more detailed discussion on SAR calculations).

Under **Component** select *mag*, *phase* or *instantaneous* from the list, and one or more of the Cartesian components x, y and z. **Currently WinFEKO can only extract iso-surfaces for near fields calculated in the Cartesian coordinate system.**

With *E_field* and *mag* selected, $\sqrt{|E_x|^2 + |E_y|^2 + |E_z|^2}$ is plotted — or any combination of the three components depending on the choice of Cartesian components. With *phase* selected, the phase (in degrees) of any **one** of the three Cartesian components is displayed (e.g. phase(E_x)). With *instantaneous* selected the real field vector is calculated as:

$$\vec{e}(t) = Re(E * e^{j\omega t})$$

and the iso-surface quantity $\sqrt{e_x^2(t) + e_y^2(t) + e_z^2(t)}$ is displayed — or any combination of the three components depending on the choice of Cartesian components.

With *H_field* the same applies as with *E_field*.

With S as quantity, the time averaged Pointing vector is calculated as:

$$\vec{S} = \tfrac{1}{2} Re(\vec{E} \times \vec{H}^*)$$

With *mag* selected, $\sqrt{|S_x|^2 + |S_y|^2 + |S_z|^2}$ is plotted — or any combination of the three components depending on the choice of Cartesian components. With *phase* selected, the phase of $\vec{E} \times \vec{H}^*$(in degrees) of any **one** of the three Cartesian components is displayed. With *instantaneous* selected the real instantaneous power density is calculated as:

$$\vec{s}(t) = \vec{e}(t) \times \vec{h}(t)$$

and the iso-surface quantity $\sqrt{s_x^2(t) + s_y^2(t) + s_z^2(t)}$ is displayed — or any combination of the three components depending on the choice of Cartesian components.

Select *Linear* or *dB* scaling under the **Scaling** options. *Min* (minimum) and *Max* (maximum) slide bars and edit boxes are available to clip the limits of the data. In the case of iso-surface display these clipping slide bars only change the colour of the iso-surface.

Also under **Scaling** is the *wt* option. By default, *wt = 0* is selected. With *instantaneous* selected under **Component**, *wt* will become enabled and the required time-instant value (in degrees) can be selected. Check the *Multiply data with* (linear scaling) or *Add to data* (dB scaling) box and enter a value to scale the data as required. (One may, for example, add the negative of the maximum dB value to normalise a graph to 0 dB.)

SAR calculations from near field data

Select the *More* button for more near field options:
The only extra option applicable to iso-surface displays is the **SAR options**. Thus **Transparency**, **Lines on surface plot**, **Contour options** and **Other options** do NOT affect the iso-surface display.

For SAR calculations the correct material parameters must be set. Enter the medium density in the edit field. (The conductivity is read from the `*.out` file.) Note that one cannot use the old format near fields to extract SAR.

The SAR will be calculated from

$$SAR = \frac{1}{2}\frac{\sigma|E|^2}{\rho}$$

With SAR selected as near field quantity, the *Maximum SAR* calculation options can be set. Enter the weight (in gram) over which the maximum SAR must be averaged. Select *Iso Vol.* or *Cube* as the shape over which to calculate the maximum SAR.

With *Iso Vol.* selected, an iterative process of iso-surface calculations will begin. A closed iso-surface, with weight approximately equal to the specified value will be searched for and the SAR in volume enclosed by this iso-surface will be calculated. The maximum SAR as averaged over any continuous volume of appropriate size (and weight) will be identified and displayed. The actual weight of the identified volume and the SAR as averaged over this volume will be displayed in the read only edit boxes under *Maximum SAR*.

With *Cube* selected, the 3D data block is scanned and the cube, with specified volume (and weight) with maximum SAR as averaged over this cube will be identified and displayed. The actual weight of the identified volume and the SAR as averaged over this cube volume will be displayed in the read only edit boxes under *Maximum SAR*.

Note: To perform such a SAR calculation a uniform (equal increments in all three directions) Cartesian 3D grid with near field data must be available in the output file (requested by the FE card for near field calculations inside a dielectric region). The density of the 3D near field grid must be small enough that the SAR can be averaged over the request weight, for example, for a density of 1000 $[Kg/m^3]$ 1g is 1 cubic centimetre. In such a case a near field grid resolution of at least 0.25cm would be required.

Unselect *Cube* AND *Iso Vol.* to calculate and display a regular iso-surface without performing iterative SAR averaged routines.

Hit the *Back* button to return to the main **Near field options** panel.

The iso-surface and model geometry can be made visible (invisible) by selecting (unselecting) the *Near fields* and *Geometry* options in the **Main display options** panel.

3.5.5.10 Near fields → Ortho-slice

On selection of this item the near field data is loaded from the current project output file and the **Near field options** panel in WinFEKO is activated. The near field data is displayed in 3D, together with the model geometry. Right-click on the near field ortho-slice to display information associated with the fields (at the point of selection).

Note: See the comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

Note further: If the checkbox next to the **Near field options** caption is checked then the panel is "activated". This means that any changes made to the options available on the panel will be applied immediately. This is useful if, for example, ortho-slices at

various levels are to be displayed without hitting the "Apply" button each time the level is changed. With the panel "activated" rendering can become very slow.

Select a frequency and near field block from the lists under **Data block selection**.

Select the *Lines*, *Surface*, *Colour* or *Arrows* options under **Display options**. The *Colour* option is only available if the *Surface* option has been unselected. With *Lines* selected, contour lines are drawn. With *Surface* selected, a colour shaded picture of the ortho-slices is displayed. Unselect *Surface* and select *Colour* to display a colour coded contour plot. With *Arrows* selected the Arrow length edit box is activated. The maximum arrow length can be set here by the user. The default is based on the near field grid spacing. To return to the default value enter 0 in the Arrow length edit box and hit the Enter key.

Select the plane (xy, yz or xz) of the ortho-slice and the level in this plane (3D Level slider) under **Plane**. For near fields calculated on other coordinate systems, e.g. cylindrical or spherical, the plane selection options would change to $rho - phi$, $phi - z$, $rho - z$ and $rho - theta$, $theta - phi$, $rho - phi$ respectively. Cartesian, Cylindrical (around z-axis), Spherical, Cylindrical (around x-axis) and Cylindrical (around y-axis) are displayed correctly in WinFEKO. Fields calculated on a Conical coordinate system or at specified points (as is possible with the FE card in FEKO) cannot be displayed in WinFEKO. For this, use GraphFEKO to extract the near field data.

The available **Near field quantity** and **Component** can also be selected if available. The Pointing vector (S) is only available when both E and H-fields have been calculated at the same positions (with FE card). SAR is only available if the near electric fields in dielectric regions have been calculated. (See section 3.5.5.9 for a discussion on what exactly is displayed when different near field quantities, components, etc. are selected.)

Select *Linear* or *dB* scaling under the **Scaling** options. *Min* (minimum) and *Max* (maximum) slide bars and edit boxes are available to clip the limits of the data.

Also under **Scaling** is the *wt* option. By default, *wt = 0* is selected. With *instantaneous* selected under **Component**, *wt* will become enabled and the required time-instant value (in degrees) can be selected. Check the *Multiply data with* (linear scaling) or *Add to data* (dB scaling) box and enter a value to scale the data as required. (One may, for example, add the negative of the maximum dB value to normalise a graph to 0 dB or scale fields to mV/m.) Check the *Fixed min/max* field to ensure that the minimum and maximum "scaling" values will NOT be changed automatically for the new data. This is useful when, for example, ortho-slices at different levels are displayed and one wants to compare these for a fixed minimum and maximum.

Select the *More* button for more near field options:
Use the checkbox and / or slide bar to set the **Transparency** options applicable to the ortho-slice **Surface** display.

Under **Lines on surface plot** select *Contour* for the default contour line display or *Grid* for a 2D grid line display. Select a height (slide bar) to associate a height, in the direction perpendicular to the ortho-slice, with the amplitude of the quantity.

Select the number of contour lines under the **Contour options**. The default is 20.

For SAR calculations the correct material parameters must be set. Enter the medium density in the edit field. (The conductivity is read from the `*.out` file.) Note that one cannot use the old format near fields to extract SAR. The SAR will be calculated as discussed in section 3.5.5.9.

Maximum SAR calculation options do NOT have an effect on ortho-slice, contour or grid displays of the near fields. (See section 3.5.5.9 for a discussion on *Maximum SAR* calculations using iso-surface.)

Hit the *Back* button to return to the main **Near field options** panel.

The ortho-slices and model geometry can be made visible (invisible) by selecting (unselecting) the *Ortho slices* and *Geometry* options in the **Main display options** panel.

3.5.5.11 Near fields → Animation

For vector arrow and colour ortho-slice plots of near fields, animation is available. First select and plot the required ortho-slice, scaling, etc. (see 3.5.5.10). With *instantaneous* selected under **Component** the *Animation* button on the **Near field option** panel will become active. Click the *Animation* button.

The **Animation control** panel is activated. This is a modal panel and the **Close** button must be clicked before control is given back to the main WinFEKO window.

Decide on the animation format by selecting either Gif or Bmp under the *Format* option.

Gif Animation: Gif animation takes longer to prepare but uses less resources (memory) than Bmp animation. Animated gifs can be saved to be viewed in most browsers and some viewers. With Gif animation the number of steps, speed and continuous (or not continuous) settings must be decided on before preparing the animation.

Bmp Animation: The preparation of Bmp animation is faster but uses more resources (memory) than gif animation. Animated bmps can not be saved and can be viewed in WinFEKO only. With Bmp animation the number of steps must be decided on before preparing the animation but the speed and continuous (or not continuous) settings can be modified while viewing the animation.

Under **Settings** select the number of animation steps. Each step will represent one value of ωt, with ωt varying between 0 and 360 degrees.

Note: Each animation step records a bitmap picture of the screen in memory. This process is memory intensive with each bitmap typically requiring between 1 and 2 MByte of RAM. With gif animation each bitmap is converted to a gif, reducing the required resources but increasing the time it takes to prepare the animation.

Select the animation speed and *Continuous animation* to display the animation steps repeatedly.

Hit the *Prepare animation* button. For large models and data sets, the preparation of the animation pictures can take a while (see progress bar). To stop the process hit the *Stop* button. The process would not stop immediately, but only after the next screen capturing has taken place.

Note: The render screen should be kept visible at all times while the animation is being prepared. Don't open or activate other application that would partially or completely cover the render screen during this period.

On completion, hit the *Animation* button to display the recorded bitmap pictures. Hit the *Stop* button to stop the animation display.

For Bmp animation, the animation speed and *Continuous animation* selection can be changed after the animation preparation has been completed. If the animation steps are changed, the *Prepare animation* button must be clicked again to prepare the new set of animation pictures. For Bmp animation, one can also manually flip through the recorded bitmap pictures by changing the ωt in the spin box after the animation preparation.

For Gif animation, the animated gif can be saved to file by hitting the *Save animated gif* button.

Click **Close** to close **Animation control**. This will also release the memory allocated by the bitmap pictures.

3.5.5.12 Currents

On selection of this item the surface and line current data is loaded from the current project output file if available (the OS card must have been used) and the **Current display options** panel is activated. A colour coded display of the surface and line currents is displayed on the model geometry. Right-click on the current display to view information associated with the currents (at the point of selection) on the rendering scene.

Note: See the comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

Select a frequency and current block from the lists under **Data block selection**.

Select the *Arrows*, *Colour* or *Smooth* options under **Current display**. The *Smooth* option results in a colour coded display averaged over the surface elements. With *Arrows* selected the *in colour* checkbox and *Arrow length* edit box is activated. The maximum arrow length can be set here by the user. The default is obtained from the segment and/or triangle side lengths of the model. To return to the default value enter 0 in the *Arrow length* edit box and hit the Enter key.

Also under **Current display** select *mag* or *instantaneous* from the drop down list. With *mag* selected, $\sqrt{|J_x|^2 + |J_y|^2 + |J_z|^2}$ is plotted. With *instantaneous* selected the real current vector is calculated as:

$$\vec{j}(t) = \mathrm{Re}(\vec{J} * e^{j\omega t})$$

By default, *wt = 0* is selected. With *instantaneous* selected, *wt* will become enabled and the required time-instant value (in degrees) can be selected.

Select *Linear* or *Log* scaling under the **Line currents [A]** or **Surface currents [A/m]** options. *Min* (minimum) and *Max* (maximum) slide bars and edit boxes are available to clip the limits of the surface and/or line current data.

The surface and line currents, as well as the model geometry can be made visible (or invisible) by selecting (or unselecting) the *Currents* and *Geometry* options in the **Main display options** panel.

Animation:

For arrow, surface and line plots of currents, animation is available. First select and plot the required current display, scaling, etc. With *instantaneous* selected under **Current display** the *Animation* button (next to Surface currents scaling) will become active. Click the *Animation* button.

The **Animation control** panel is activated. This is a modal panel and the **Close** button must be clicked before control is given back to the main WinFEKO window. For detailed information on the options and settings on the **Animation control** panel see section 3.5.5.11.

3.5.5.13 UTD rays

Select this item if a FEKO problem involving UTD regions has been solved — *with the option to save the ray file*. On selection the data in the `*.ray` file is loaded (if available — see UT card, section 8.2.38) and the **Ray display options** panel is activated. The rays data is displayed on the model geometry.

Note 1: Ray files can quickly become very big which will make the display of the rays very slow. If more than 4000 ray lines are identified in the `*.ray` file a warning appears. Select *Yes* to continue loading all ray data and *No* to display only the first 4000 rays (data already loaded) and disregard the rest of the data in the `*.ray` file.

Note 2: See comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

Select (or unselect) the *Display ray file info* to switch the display of the `*.ray` file information on (or off). Click the *Clear ray file* button to release the memory (and associated data) allocated for the ray file information.

Under **Visibility** select the ray information that should be displayed.

One can select the type of rays to display and determine the maximum number of reflections in the *Ray type* group. If, for example, *Corner diffraction* is unchecked such rays will not be displayed. Note that *Diffraction* only refers to single diffraction — if this is unchecked double diffracted rays will still be shown unless the *Double diffraction* field is also unchecked. Rays with more than the specified number of reflections will not be shown.

Each ray group includes all rays that start and end at the same point. The **Selection** options can be used to display individual ray groups. Check the *Ray groups* checked box to display only the ray group with the number selected from the drop down list. If the *Ray group* box is unchecked all rays are displayed simultaneously — this may require a very significant rendering time. The *Ray number* field can be used to view the rays in the group one by one. The *Far field crop* slide bar can be used to restrict far field points to within the geometry display.

The ray data is also considered when calculating the model extent (used for model normalisation). It could thus be necessary to re-normalise the geometrical model display after clearing the ray file information.

3.5.5.14 Clear data

This item "clears" all information associated with the output data of the current project. All memory allocated when loading the output file data is released. Unlike the *Clear* item under Preprocessing, this command does not clear / release the FEK or NEU file model data.

3.5.5.15 Solution information

Not available in WinFEKO yet. See beginning and end of the output file for solution info associated with memory usage and solution times.

3.5.6 Display menu

Used to set various options for the display of the geometric model.

3.5.6.1 Geometry → Main display options

The *Main display options* panel is the default active panel when WinFEKO is loaded. Here the most often used model display options can be selected. With this panel active, all WinFEKO hot-keys can be used (see section 3.3).

Under **Visibility** the following options are available:

- *Axis-arrows:* Toggle axis display.
- *Axis-small:* Toggle display of small axis in upper right corner of render screen.
- *Axis-grid:* Toggle axis grid values display.
- *Cutplane:* Activate cutplane options (see also section 3.5.6.5).
- *Elt. direction:* When selected, the back of surface elements will be displayed in a different colour shade than the front. Front and back are defined according to the right-hand-rule.

- *Elt number:* When selected, the number associated with each segment, element, polygon or cuboid will be displayed. This will slow down the model display and rendering considerably with large models.
- *Fast rotation:* Select this to render a boundary box around the model on dynamical rotation, panning and zooming (as well as on zoom-to-window). This is very useful with large models.
- *Lead lines:* Add lead lines to entity numbers when *Element Info* is selected.
- *Legend:* Display a legend. The type of legend that will be displayed depends on other selections (e.g. with surface and segment currents loaded, a legend associated with the colour coding will be displayed).
- *Node labels:* When selected the FEKO DP card node labels will be displayed.
- *Normal vectors:* Normal vectors associated with triangular and polygonal element directions will be displayed. The direction of the normal vector is defined according to the right-hand-rule.

Under **Display** the following options are available:

- *Geometry:* When selected the geometry associated with the current project (`*.fek` or `*.neu` files) is displayed (if the geometric data has already been loaded). Unselect this option to hide (not delete) the model geometry display.
- *Excitation:* When selected the excitation associated with the model is displayed (if the geometric data has already been loaded). Unselect this option to hide (not delete) the excitation display.
- *Currents:* When selected the segment and surface currents obtained from the FEKO output file are displayed (if the current data has already been loaded). Unselect this option to hide (not delete) the segment and surface currents display.
- *3D patterns:* When selected the 3D far field patterns obtained from the FEKO output file are displayed (if the far field data has already been loaded). Unselect this option to hide (not delete) the 3D far field display.
- *Near field:* When selected the near field ortho-slices and iso-surfaces obtained from the FEKO output file are displayed (if the near field data has already been loaded). Unselect this option to hide (not delete) the near field display.
- *Infinite plane:* When selected, the infinite plane(s) associated with the current project is displayed. This could be the ground plane as defined using the BO card or the ground plane (and/or infinite layers) associated with the Multilayered substrates (GF card).

- *Request fields:* When selected the requested fields as specified with the FE and FF cards in the `*.pre` file are displayed (if the request field data has already been loaded). Unselect this option to hide (not delete) the requested fields display.

- *Transmis. lines:* If the model contains TL cards, they are displayed by checking this field. The lines are drawn in yellow between the segments they are connected to. The lines are not drawn from the centres (where they are physically connected) such that one can determine the polarity of the connection.

Under **Wire segments** select:

- *Lines:* Display segments as lines.

- *Surface:* Display segments as tubes. Unselect this when the model has a large number of line segments to avoid slow rendering — especially when printing to vector formats.

Under **Triangles** select:

- *Lines:* Outlines of the triangles are displayed.

- *Surface:* The triangle surfaces are displayed. With *Flat* selected a true representation of the triangulated geometry is displayed. *Background* removes any colours from the display of the surfaces of the triangles. *Smoothed* represents a smoothed display of the triangulated geometry.

- *More:* If the model contains anisotropic dielectric layers (specified with the SK card), the *More* button becomes visible. This opens a new panel where one may select an SK card and the layer number to show the fibre (principle) direction of the layer.

Under **Cuboids** select:

- *Lines:* Outlines of the cuboids are displayed.

- *Surface:* Cuboid surfaces are displayed.

Under **Polygons** select:

- *Lines:* Outlines of the UTD polygons are displayed.

- *Surface:* Polygons surfaces are displayed.

3.5.6.2 Geometry → Previous display options

Select this item to go back to the previously **active** display options panel. It would be more efficient to use the speed button associated with this command (see section 3.4.3.)

3.5.6.3 Geometry → FEK display options

On selection of this item the **FEK file display options** panel is activated. Labels associated with the `*.fek` file are presented in the *Available Labels* list.

Note: See comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

To display only selected labels, copy the appropriate labels to the *Selected Labels* list (use the arrow buttons). Select the *Selected labels* option under **Label visibility** to display entities associated with the selected labels.

Unselect any of the options under **Element visibility** to hide specific types of entities. These options work together with the options available in the **Main display options** panel.

Select one of the options under **Colours** to determine how colours are used with the model display. Select the *Legend* option in the **Main display options** panel for colour code identification.

The options are:

- *Element type:* Use colour codes associated with wire segments, metallic triangles, dielectric triangles, cuboids and UTD polygons. This item is the default colour selection.
- *EM properties:* Display entities associated with specific EM properties in different colours. For example, perfectly conducting metallic segments and loaded segments will be displayed in different colours.
- *Label number:* Display the different `*.fek` file labels in different colours.
- *Medium:* Display the different media in models using the surface equivalence principle.

Select the *More* button (top-right) to activate a panel with more **FEK file display options**.

Under **Multi-layer substrates** select *Enlarged substrates* to draw the substrates at 10 times its actual thickness. Select *Draw lines* to draw outlines on the substrates. Select *Show substrate no* and then enter the numbers of the substrates that should be displayed in the adjacent edit box. For example, to hide substrate 2 and display substrates 1 and 3 type 1,3 in the edit box.

Select the *Back* button to return to the main **FEK file display options** panel.

3.5.6.4 Geometry → NEU display options

On selection of this item the **Neutral file display options** panel is activated. Layers associated with the NEU file are presented in the *Available layers* list, *if a neutral file has been loaded for display* (see section 3.5.2.2).

Note: See the comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

To display only selected layers, copy the appropriate layers to the *Selected Layers* list (use the arrow buttons). Select the *Selected layers* option under **Layer visibility** to display entities associated with the selected layers.

Unselect any of the options under **Element visibility** to hide specific types of entities.

Select one of the options under **Colours** to determine how colours are used with the model display. Select the *Legend* option in the **Main display options** panel for colour code identification. The options are:

- *Element type:* Use colour codes associated with line segments or metallic triangles (cuboids and polygons not implemented yet). This item is the default colour selections.
- *Layer number:* Display the different NEU file layers in different colours.

3.5.6.5 Geometry → Cutplane options

The **Cutplane options** panel is activated when this item is selected. Cutplanes can be defined to hide selected regions of the model or data.

Note: See the comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

Under **X=constant**, **Y=constant** and **Z=constant** set the position of the cutplane in the available edit box. For example, under **Z=constant** if the constant value is set to 0, then only geometries in the region $Z <= 0$ will be displayed. Use the slide bars to select different cutplane positions. Select *Active* to make the cutplane active. Select *Reverse* to reverse the region that will be "cut out". For example, with *Reverse* selected under **Z=constant** (constant values set to 0) only geometries in the region $Z > 0$ will be displayed.

Under **Entities to cut** select the entities on which the active cutplanes must operate. It is for example possible to use a cutplane on metallic triangles while wire segments are displayed in the entire region (*Wire segment* unselected). Also, result data can be left uncut with the geometry cut at the selected plane. But currently the *Result data* option only has an effect on near field data and not far field radiation patterns.

Remember that the *Cutplane* option in the **Main display panel** must be selected for any of the cutplane options to work. If you make any modification on the **Cutplane options** panel and hit the *Apply* button, then the *Cutplane* option in the **Main display panel** will be set automatically.

3.5.6.6 Geometry → Advanced visibility

The **Advanced visibility** panel is activated when this item is selected.

Note: See the comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

The *Size of dipole points* controls the size of point dipole sources.

The *Shrink cells* as well as *Enlarge radius* options are available for **Segments** and **Impressed Currents**.

Use the appropriate slide bars to set the length of the segment display. When set to 100% each line segment is displayed with its full length. The default is 90%. The 90% setting makes the individual line segment identification easier.

Use the *Enlarge radius* slide bars to enlarge the radius of the segment or impressed current display. This is an artificial enlargement used for display purposes only and has no effect on the solution. When set to 1 each line segment is displayed with its actual radius as defined by the user (this is the default). When set to between 2 and 100 each line segment is displayed with its actual radius enlarged by the "enlarge factor". This is useful when visualising thin wire segments and impressed current line elements.

3.5.6.7 Setup → Requested fields

Used to view the positions at which near and far fields will be calculated, as requested by the user with the FE and FF cards.

Note: See the comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

Note further: If the checkbox next to the **Requested Fields** caption is checked then the panel is "activated". This means that any changes made to the options available on the panel will be applied immediately. This is sometimes useful but with the panel "activated" rendering can become slow.

Select the block to display using the *Near Fields* and *Far Fields* drop down lists. The corresponding request field "type" will be displayed in the black information box below the drop down list. Select the colour box to change the colour of the requested near and far field display.

Select *Points*, *Spheres*, *Lines* or *Surfaces* as options to display the selected fields. *Points* usually works well but are sometimes difficult to see when only a few request field points are present. *Spheres* would show a few points more clearly, but for a large number of field points *Spheres* render very slowly. *Lines* and *Surfaces* are only available as options for far fields and near fields in a Cartesian coordinate system.

Deactivate requested far and near field display with the *Request field* checkbox on the **Main Display Options** panel.

3.5.6.8 Setup → Excitation

This item allows the user to selectively display sources. The *Set number* groups sources that are active at the same time, i.e. it is incremented for each source which has the "New source" flag set. The *Source number* is the number of the source in the current set. Note that the AP card results in a large number of dipoles (A5 and A6 cards) such that the complete card can only be displayed with *Source number* set to `All`. *Display types* is used to turn off the display for certain excitation types.

If only one source is selected, the amplitude and phase of the source is displayed when the *Apply* button is clicked. As always, if the box next to *Excitation* at the top is checked, changes are processed immediately.

3.5.7 Tools menu

Some tools are available in WinFEKO to assist with model display and trouble shooting when the FEKO solver gives warnings or errors.

3.5.7.1 Find → Element

The *Find elements* panel is activated when this item is selected. The commands in this panel operate similar on both NEU and FEK models.

Note: See comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

The range of elements is presented under **Valid range**. This can be quite useful with NEU files (neutral file models) where numbering does not necessarily start at 1.

Type in an element number (or list of element numbers) to "find". As an example, type: `1,2,5,7-10` to search for element numbers 1,2,5,7,8,9 and 10.

Under **Options** select one or more of the following:

- *Display elements:* Enable or disable the display of the selected elements. This option will thus override all the other options available.
- *White colour:* If selected the elements in the *Find list* will be highlighted.
- *Lead line:* Add a lead line between the selected element and the selected element number.
- *Show numbers:* Display the numbers associated with the selected elements.
- *Both sides:* Draw lead lines, highlighting and numbers on both sides of the selected elements. (Not applicable to line segments).

Click the *Entity Selection* button to jump the the *Entity Selection* panel where the on-screen entity selection (or picking) options can be set (see section 3.5.7.4).

3.5.7.2 Transformation

The *Transformation control* panel is activated when this item is selected.

Note: See comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

Note further: If the checkbox next to the **Transformation control** caption is checked then the panel is "activated". This means that any changes made to the options available on the panel will be applied immediately. This is sometimes useful but with the panel "activated" rendering can become very slow.

Under **Vantage points** the require view angle can be set. The ϑ and φ angles describe the direction (in spherical coordinates) of the viewpoint relative to the origin.

Under **Zoom** the user may set the *Zoom factor* and the *Zoom step %* which determines the rate by which the zoom factor increases or decreases when, for example, the *Zoom in* or *Zoom out* buttons are clicked.

Under **Translation** the required pan position of the model can be set.

The origin of rotation can be changed under the **Origin** options. By default, the model is rotated around the origin of the Cartesian coordinate system (x=0, y=0 and z=0). Enter the Cartesian coordinate point around which the model must be rotated in the appropriate x, y and z edit boxes (in metres). If a vertex or geometry entity is selected, the *Selected Entity* button will become enabled. Click it to fill the x, y and z edit boxes with the coordinates of the selected entity. Click the *Model Centre* button to fill the x, y and z edit boxes with the coordinates at the centre of the model under display.

3.5.7.3 Render test

Select this menu item to open the **Render Test** window. This window display information about the OpenGL rendering hardware and software on the system at hand. Click the *Start Test* button to start the render test for render speed comparison to other systems.

3.5.7.4 Entity Selection

With this menu item any of *Geometry*, *Vertex*, *Near Fields*, *Far Fields* or *Current* entity selection can be activated. The **Entity Selection** panel is enabled.

Note: See comments on the *Apply*, *OK* and *Cancel* buttons in section 3.5.5.7.

Under *Picking Display Options* select the way the "picked" geometry element will be displayed on screen. This is similar to the display options discussed in section 3.5.7.1.

Under *Picking Options* select the type of entity that must be highlighted / displayed when clicking on the model and/or data on the display screen. To pick an entity the user must **right**-click on the appropriate geometry or data object.

For far field, near field and currents picking, the corresponding data must have been loaded and displayed in the model view window. For geometry and vertex picking, a geometry model must be visible in the model view window.

Note: With the Far Field, Near Field and Currents Option panels visible, picking for the corresponding far field, near field and current objects are automatically selected under the *Picking Options*.

Right-click anywhere on the background of the model view window, i.e. not on any geometry or data object, to "erase" the displayed information enabled by picking.

3.5.8 Options menu

3.5.8.1 General settings

Here a number of general program options can be set. All of these options can be saved to a configuration file. The default configuration file is `winfeko.cfg`. If this file is present in the FEKO home directory, these settings will be the default settings that WinFEKO starts with. It is however important to note that some of these settings are also saved in the project file `*.wfp` associated with each WinFEKO project. In other words, on startup, WinFEKO loads the settings from the `winfeko.cfg` file. When a project is loaded, the `*.wfp` file is loaded (if it exists) and some general settings associated with the specific project overrides the default WinFEKO general settings.

To change the default WinFEKO configuration file, make the appropriate changes under all **General settings** tabs. Then click the *Save* button and save it as `winfeko.cfg` in the FEKO home directory. The general settings can also be saved in any other configuration file for later use. Such settings can be loaded by clicking the *Load* button and selecting the appropriate configuration file.

To save the settings associated with a WinFEKO project, make the appropriate changes under all **General settings** tabs, hit the *Apply* or *OK* button and save the project.

The different general settings options will now be described:

1. **Display:** Under *Element info / Offset Direction* select *Normal* to display element information at a fixed 3D position next to the element (e.g. with triangles, this position is in the direction of the normal vector associated with the triangle). Select *View* to display element information always in the direction of the user's view point. This last option is useful when element information is hidden behind geometric elements in the model.

   Under *Request Field Info* select the colour in which the requested near and far fields must be displayed on the screen. This is not the result data, but the positions of the requested fields as defined by the FE and FF cards.

   Under *Axis Length* set the length of the axis. With this values set to zero, a default axis length will be calculated and used by WinFEKO.

   The AP card is converted to dipole sources. These are normally rendered as arrows which may require a large amount of RAM and rendering time. For such models one might uncheck *Display arrows* to view these sources as short lines only.

2. **Rendering:** Under *Dynamic rendering* select *Render surfaces* to render all surfaces when actively rotating, translating, zooming, etc. models. If this is not selected the model will be displayed correctly, but only representative lines will be visible when the model is manoeuvred. For medium and large sized problems, it is best to unselect the *Rendering surfaces* option.

3. **Saving:** Under *Project saving* select *Prompt on close* to ensure that the user is warned when the project information has not been saved to the `*.wfp` file. Remember, it is not critical to save a project in WinFEKO because the `*.wfp` file do NOT contain the critical information associated with the FEKO input file (`*.pre` file). Select *Window settings* to have WinFEKO startup with the same window size (width / height) as the previous time the program was run. If not selected, WinFEKO will start with the default size window (which is slightly smaller than the users screen size).

4. **Reading:** Under *Out file reading* select *Report warnings* to have WinFEKO report any WARNINGS present in the `*.out` file when the result data is loaded.
   Read DP cards can be unchecked on slow machines for models which contain a large number of DP cards. This may speed up the reading of the `*.fek` file, but it is no longer as critical as with the older versions of WinFEKO or older `*.fek` files.

5. **Other:** Under *Batch files* select *Call InitFEKO* to explicitly add a `call initfeko.bat` line every time PREFEKO or FEKO is run from within WinFEKO. This is only useful on systems where FEKO has been installed and the appropriate path and environment variables could not be set due to file write restriction (for example, when changes to the `autoexec.bat` file is not allowed on a system).

   Under *Default Selection Option* select the default option that must be used when picking a model entity.

   Under *FEMAP version* select the version of FEMAP (if any) installed on your system. This is necessary for WinFEKO to create the correct FEMAP `*.mod` file when a new WinFEKO project is created. If *None* is selected here, WinFEKO will not create a FEMAP `*.mod` file at all.

   Under *3rd Party CAD String* select the path and string, as well as extension, of a third party CAD program that must be executed (instead of FEMAP) with the FEMAP execute button. The specified extension is added to the current project name and this is passed as parameter to the specified executable. This third party CAD program will be executed only if the *Active* checkbox has been selected. (It is possible to replace one of the default FEMAP model files with a model file for the specified CAD program — these are located in the `Defaults\FEMAP` subdirectory of the FEKO installation.)

Note: Remember to hit the *Apply* or *OK* button for the changes to take effect, and the *Cancel* button to disregard all changes.

3.5.8.2 Toolbars

Unselect (or select) the *Main* item to hide (or display) the *Main* toolbar. Unselect (or select) the *Display* item to hide (or display) the *Display* toolbar. Unselect (or select) the *Render* item to hide (or display) the *Render* toolbar panel.

3.5.8.3 Command line parameters This menu is used to specify optional command line parameters to PREFEKO and FEKO. This is mostly useful for debugging purposes. Note that these parameters are saved in the `winfeko.ini` file and one should remove them once they are no longer required.

3.5.9 Help menu

3.5.9.1 User's manual

Select the Users Manual item under Help to start Acrobat Reader and automatically load the general FEKO *User's Manual* — available in PDF format. If nothing happens when this item is selected, then a PDF viewer has probably not been installed on your system. Acrobat Reader is available for installation on the FEKO CD.

3.5.9.2 About WinFEKO

Select this item to get the latest WinFEKO version number and development information.

4 The editor EditFEKO

A PREFEKO input file is a standard ASCII text file that may be created with any available text editor. The model geometry and desired calculation are entered through lines of text, referred to as cards. Each card may have a number of parameters which must appear in fixed column positions. The editor EditFEKO was designed to simplify the process of generating `*.pre` files. It is a basic text editor with customised functionality. The main functionality is the specialised card editor for PREFEKO input cards. It also has limited ability to handle `*.opt` OPTFEKO input files, but currently uses a plain text editor for `*.tim` TIMEFEKO input files.

The program may be started from a command prompt by entering

```
editfeko example.pre
```

where the optional parameter (here `example.pre`) specifies the file to open. If the parameter does not contain an extension, the extension `pre` is added. Also, if such a file does not exist, it will be created. EditFEKO can also be opened from inside WinFEKO.

The interface consists of an editor area where more than one file may be opened. In the standard PREFEKO mode the button panel (which may be turned off, or moved to either side of the editor area) provides quick access to the card editors. The status bar on the bottom of the window contains the line and column numbers, indicates whether "Insert" or "Overwrite" mode is on, and also gives an indication of the file modification status.

Note:
The editor is designed for a system with "Small fonts" selected in the "Screen settings" property of Windows. If "Large fonts" is selected some items may be misaligned.

4.1 General

4.1.1 File menu

The *File* menu allows opening and closing files and is similar to standard Windows applications. The *New* sub menu item opens a sub menu where the user may choose **.pre* for PREFEKO, **.opt* for OPTFEKO, or **.tim* for TIMEFEKO files. EditFEKO will enter the appropriate mode depending on the selected option.

The *Save* menu will ask for a filename if the file does not have one, otherwise it will save (without confirmation) using the current filename. The *Save as UNIX* menu can be used to convert to UNIX end-of-line characters. *Save all files* saves all unsaved open files.

This is followed by a list of recently used files and the printing and exiting menu items. Various "hot-keys" are presented next to the respective menu items. Note that <Alt><X> will save the current file and quit. If there are other unsaved files open, the user will be given the option to save them.

4.1.2 Options menu

This menu allows some customisation of EditFEKO. Both the screen and printer fonts may be set — it is recommended that a fixed width font such as "Courier New" be used. The *Options* menu also allows the user to move the button panel to the left or right of the editor area, or to switch it off completely allowing more space for the editor windows.

The *Options* menu may also be used to enter the superuser mode. This will print a warning, but is open to anybody using EditFEKO. This mode activates certain options which are only available in the superuser mode of FEKO. The superuser mode in FEKO is only used during program development and is not available to the general user.[3] Thus it is not recommended to use EditFEKO in the superuser mode. If the user tries to edit an existing card containing superuser parameters, EditFEKO will also prompt the user and switch to superuser mode.

The *FEMFEKO* mode is only useful for developers working on the FEKO/FEM hybrid.

If the *Sort drop down list* menu item is selected, the contents of suggestion boxes (the response to clicking the right mouse button over an an input field dialog box), are sorted alphabetically. If not, they appear in the order they are created. See also section 4.2.2.

The OPTFEKO editor mode will accept both English and German keywords, and it will retain the existing language when editing blocks. The *Options* menu allows the user to select the language for newly created control blocks in the `*.opt` file.

4.1.3 Window menu

This menu presents the standard Windows functionality to arrange or select the editor windows on the main editor area.

4.1.4 Help menu

This menu presents the version of EditFEKO (*About*) and presents a shortcut to the *User's Manual*. The shortcut requires that some or other PDF viewer (such as the Adobe Acrobat reader) is associated with `*.pdf` files.

4.2 PREFEKO mode

4.2.1 Generating input cards

Cards can be entered and edited directly in the editor window. In most cases the user would call one of the card editors by clicking on the appropriate button on the button

[3] Certain options and functions are only accurate and / or efficient for specific problems and require an intimate knowledge of the code.

panel or selecting from the *Geometry cards* and *Control cards* menus. The buttons are grouped according to functionality namely

- Definition of node points, labels, segmentation parameters and variables (# button).
- Cards to generate surfaces which will be meshed into triangles elements.
- Cards generating wire segments.
- Cards for generating dielectric / magnetic volumes to be subdivided into cuboids, or specification of dielectric regions.
- Cards used in connection with the PO formulation.
- Cards used in connection with the UTD formulation.
- Modifications to the geometry, such as translation, scaling and mirroring.
- The EG card signifying the end of the geometry. All the cards whose buttons are above this button should be before it in the input file, and those below must be after. The IN card, IF–ELSE block cards and FOR–NEXT loop cards appear next to the EG card as they can be used both above and below it.
- Cards dealing with finite medium properties and finite grounds.
- Cards dealing with loading of wire segments.
- Cards to specify the excitation type, frequency and power of the excitation.
- Cards used to control which data is written to the output files.
- Cards to specify where to calculate the fields.
- The EN card which signifies the end of the input file.

When the card editor is selected from either the button panel of the menu, there is an option to *Add card*. This adds the card without closing this editor. This is useful when defining a number of similar cards, for example, when specifying the first group of DP cards at the start of a model. Often only one or two parameters differ between cards. Pressing <Enter> has the same effect as clicking on the *Add card* button. Similarly <Esc> closes the card editor as if the *Cancel* button has been clicked.

In the card editor, moving the mouse pointer over the labels may give more information of the specific input field. If the user is uncertain about the meaning or units of a given field, it is always advisable to move the mouse pointer to the label to determine if any additional information is available.

All the input fields in FEKO have a fixed length. In the card editor it is not possible to enter longer strings than allowed for the particular field. Note, however, that spaces in the input field may be overwritten by new keys when the input field has maximum length. If the cursor is at the end of the field or at any character other than a space, no new keys are excepted once the maximum length is reached. Then it is necessary to delete characters before any additional ones can be entered.

Editing an existing line

The card editor can also edit existing lines. The line at the current cursor position may be edited either by pressing <F1>, selecting *Edit → Edit line* from the main menu, or by right clicking at any position in the editor and selecting *Edit line at cursor* from the pop-up menu. Note that right clicking does not change the cursor position. When the selection (or highlighted part) in the editor spans more than one line, the line or lines cannot be edited. This also applies for multi-line cards (such as the GF card) — in this case the cursor must be on the first line of the card and all lines will be edited simultaneously.

EditFEKO processes each card as if it does not contain errors. For some of the fields where the user may select a number of options, the card editor's display will default to a "common" value if the input field is invalid. *Thus it is advisable to close the card editor with* OK *even when it was used just to check the validity of the line.*

If the user type, for example, `PS` and then press <F1>, EditFEKO may treat it either as a new card (with default) options, or an existing card with all options blank (zero). Setting all options to zero does not always make sense and EditFEKO thus treat this case as a new card and use the normal defaults. If the user wants to have a card with all option zero, he should type the card name followed by a space before pressing <F1>.

When editing existing lines, the *Add card* button is not available as the card is edited and written back to the editor. The card editor will keep the last parameters entered (even if it is closed) until a different type of card is edited. In this case pressing <Enter> has the effect of clicking the *OK* button.

The card editor does not edit cards generated by PREFEKO (such as the DR card). The user should rather use the input cards that are read and converted by PREFEKO. Generally, it should not be necessary to use cards that are not available in EditFEKO.

It should finally be noted that, while the card editor is open, no other action is allowed. It is not possible to resize the window or hit the close button (marked [×]) on the top right corner.

4.2.2 Parameter suggestion

While using the card editor, right clicking on the input field dialog boxes presents a list of suggested values. This may be either point or element labels (depending on the card and the input field) for the integer parameters, or a list of variables for the real input fields.

The list is constructed as PREFEKO would read it, i.e. only the labels or variables defined above the current line are used. They are listed in the order that they are specified by the input cards. (Unless they are sorted using the *Options → Sort drop down list* menu option. For the list of variables, the predefined variables (see page 6-5) are listed first.

If the mouse is clicked anywhere inside the list area, the text in the input field is replaced by the item under the cursor. (This highlights as the mouse moves over it.) The <Enter> key replaces the input field by the currently highlighted item. The list of suggestions may be closed by pressing the <Esc> key, or clicking outside it.

If the list is longer than 10 lines, it may be scrolled using the scroll bar or by clicking on an item and moving the mouse down (or up) outside the list border while keeping the mouse down. Releasing the mouse button places the current item in the input field.

4.2.3 Variable editor

The **#** button on the top left corner of the button panel launches the card editor to edit variables. This is useful as it presents a list of functions and operations understood by FEKO and calculates the value of the variable as it would be evaluated by PREFEKO at this point. (Note that EditFEKO does not expand FOR loops. Thus variables depending on other variables defined inside FOR loops will not be evaluated correctly.)

The functions, variables or operations may be selected from the three drop down boxes and a special group for the "FILEREAD" function. The selected item is automatically placed at the current cursor position. Note that variables and functions are highlighted after insertion, but operations are not. If a function is selected while some part of the variable string is selected, the selected text will be inserted as the argument of the new function. Variables, operators and the "FILEREAD" function replace the currently selected text.

4.2.4 Edit menu

Besides the standard Windows functionality this menu provides the *Insert file* menu to copy the contents of another file to the current cursor position. It also provides access to the card editor for the current line.

The *Edit → Undo* option will undo the last change to the editor only. Selecting it again will just undo the effect of the *Undo* command.

The items *Edit → Comment* and *Edit → Uncomment* allow the user to comment out blocks of text or remove the comments. This is often useful when debugging large files. The comments are inserted as ** followed by a space and only removed if this sequence is present.

For example, in the lines

```
** ** Select the label
** LA   1
**Next comment
```

the comment characters will not be removed from the third line using this function. It was also not generated by the *Comment* function. These commands are also available using the "hot-keys" <Alt><C> (for *Comment*) and <Alt><U> (for *Uncomment*) or by right clicking in the editor and selecting from the resulting pop-up menu.

4.2.5 Search menu

It is possible to search the editor for specific text, case sensitive or not. The search always starts from the current cursor position. If it is desired to start the search from the top of the file, the cursor must be moved there (using <Ctrl><Home>).

Note that there is also a fast card search. By right clicking on any button on the button panel, the next occurrence (starting from the current cursor position) of the associated card is found.

4.2.6 Run menu

This allows PREFEKO, the FEKO solver, WinFEKO and GraphFEKO to be run from inside EditFEKO. Note that when running PREFEKO or FEKO, the current file is *saved without confirmation*. For WinFEKO and GraphFEKO this is not the case as they do not read the `*.pre` file.

This menu also allows passing parameters to PREFEKO and FEKO (see section 6 for PREFEKO and section 7.2 for FEKO).

4.3 OPTFEKO mode

In this mode the OPTFEKO control blocks can be edited. The cursor must be on the keyword of the specific block in order to edit it with <F1>. In the block of variables, only the variables from the current line downwards will be edited — thus the cursor should be on the first line of the variable block when editing it.

The comment buttons will add a comment line regarding the particular block and in some cases also captions for the different input fields in the block.

4.4 Important keystrokes

The arrow keys as well as <Pg Up> and <Pg Dn> behave in the normal fashion. The following keys may be different to some other applications.

Cursor Movements

Move a word left	<Ctrl><Left Arrow>
Move a word right	<Ctrl><Right Arrow>
Move to top of visible page	<Ctrl><Page Up>
Move to bottom of visible page	<Ctrl><Page Down>
Move to beginning of line	<Home>
Move to beginning of line	<End>
Move to beginning of file	<Ctrl><Home>
Move to end of file	<Ctrl><End>

A block may be selected using the mouse, or pressing <Shift> and using the normal movement keys. If a block is selected, it will be overwritten when a key is pressed. The following list of "hot-keys" are often used

"**Hot-keys**"

Copy to clipboard	<Ctrl><C> or <Ctrl><Ins>
Cut (delete) to clipboard	<Ctrl><X> or <Shift><Del>
Paste (insert) from clipboard	<Ctrl><V> or <Shift><Ins>
Save	<Ctrl><S>
Save all files	<Ctrl><A>
Save and exit	<Alt><X>
Edit line	<F1>
Comment line(s)	<Alt><C>
Uncomment line(s)	<Alt><U>
Find	<Ctrl><F>
Find next	<F3>
Find and replace	<Ctrl><R>
Run PREFEKO	<Alt><2>
Run WinFEKO	<Alt><3>
Run FEKO	<Alt><4>

Right clicking with the mouse on a card panel button searches for the next occurrence of that card. Right clicking in an input dialog box presents a list of possibilities.[4]

[4]Note that only fields entered above the current cursor line are used. If, for example, the user is specifying a BL card before the first DP card, right clicking on the input fields will have no result.

5 The program GraphFEKO

The program GraphFEKO has been developed as a separate executable but it is part of the Graphical User Interface (GUI) for FEKO in a MS Windows environment. It is used for creating 2D plots of the calculation results.

5.1 Running GraphFEKO

To run GraphFEKO, select *GraphFEKO* in the FEKO folder under the Windows *Start* menu. It can also be executed with a `*.out` file as command line parameter, for example

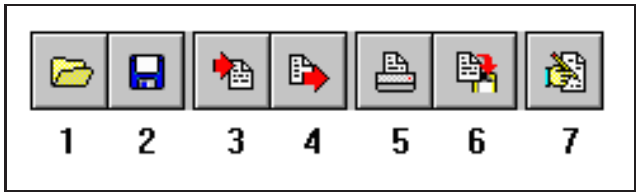
```
graphfeko.exe example_01.out
```

In such a case GraphFEKO will load the data available in the given output file. If GraphFEKO is executed from within WinFEKO the data in the output file of the current WinFEKO project will be loaded into GraphFEKO.

5.2 Toolbars in GraphFEKO

The toolbars in GraphFEKO can be organised into 2 main groups. Most buttons on these toolbars are associated with menu items and their functionality will be described in the corresponding menu item sections.

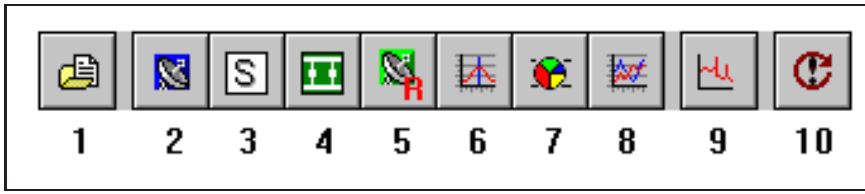
5.2.1 FILE control toolbar



Used for GraphFEKO file control and printing. The functions of the buttons are:

1. Open a graph file (see section 5.3.1.1).
2. Save a graph file (see section 5.3.1.2).
3. Load data file (see section 5.3.1.8).
4. Save data file (see section 5.3.1.9).
5. Print graph (see section 5.3.1.10).
6. Print to file (see section 5.3.1.11).
7. Main graph edit (no associated menu item). This speed button is used to return to the *Main graph settings* panel.

5.2.2 Data extraction toolbar



Used for extracting data from the FEKO output file. The functions of the buttons are:

1. Import / select a FEKO `*.out` file (see section 5.3.2.1).
2. Extract antenna parameters (see section 5.3.2.3).
3. Extract S-parameters (calculated with the SP card, see section 5.3.2.4).
4. Extract network parameters using antenna source (see section 5.3.2.5).
5. Extract receiving antenna data (see section 5.3.2.6).
6. Extract currents (see section 5.3.2.7).
7. Extract far field data (see section 5.3.2.8).
8. Extract near field data (see section 5.3.2.9).
9. Extract results for an adaptive frequency interpolation (see section 5.3.2.10).
10. Reload `*.out` file (no associated menu item). This button reloads the current `*.out` file from disk and opens the last parameter panel. It is very useful when one repeatedly changes parameters, reruns FEKO and plots the data.

5.3 Main menu structure

5.3.1 File menu

5.3.1.1 Open

Select this item to open an existing graph created with GraphFEKO. A standard windows *Open file* dialog is activated. Only files with the extension `*.wfg` can be opened.

5.3.1.2 Save

Select this item to save the graph on the active GraphFEKO window. The graph is saved to the file name associated with and displayed at the top of the active window.

5.3.1.3 Save as

The graph on the active GraphFEKO window can be saved to a different filename. A standard *Save as* dialog is activated. Select a new filename with extension `*.wfg` — if any other extension is specified it will be changed to `*.wfg` as GraphFEKO can only open files with this extension.

5.3.1.4 Close

This item closes the active GraphFEKO window. If the graph on the active GraphFEKO window has not yet been saved since the last changes (or edits), the user is warned before the window is closed.

5.3.1.5 Clear template

GraphFEKO allows the user to save graphs as templates to use when new graphs of similar type are created. These templates have very specific names which depend on the options used to create the type of graph. The *Clear template* menu allows the user to clear the template for a specific type of graph and revert to using the GraphFEKO defaults. It opens a separate window where the user can select the various options which determine the template name. As the user changes the options, the template name is shown in the box at the bottom of the form. See also section 5.3.1.7.

5.3.1.6 Load template

The templates which GraphFEKO uses when creating new plots are standard graph files with specific names. These files can be loaded and edited similar to loading and editing standard graph files. The *Load template* menu merely simplifies selecting the template for a specific type of graph. The modified template can be saved such that future graphs of this type will be created from this template. See also section 5.3.1.7.

5.3.1.7 Save template

When creating a new plot, GraphFEKO first determines if there is a template for this type of graph. The template contains information such as the caption fonts, the axis limits, the line colour and type, etc. In principle the new plot will be similar to the template, but using the current data. Any graph can be saved as a template — including a graph that has just been created. This is mostly useful when one wants to create a series of similar graphs.

The templates have very specific names which depend on the parameter under consideration and the type of data (linear/log/dB). The extension depends on the graph axis — `*.lin` for line graphs or `*.pol` for polar graphs. All the different field quantities on one parameter windows use the same template. For example, both the axial ratio and the electric field on the *Far field* panel use `FFPar.lin` when plotting a line graph with linear data. If the new plot has more lines than the template, the template lines are used repeatedly. Thus if the template has only one line all lines on the new graph will be the same w.r.t. colour, line type, markers, etc. (Note that plotting graphs from the *Current extraction* panel uses the same templates as from the *Network parameters* panel.)

If the user selects *Save template* while the panel from which the graph was created is still open, GraphFEKO will suggest the correct template name. If not the user has to

specify the type of graph by selecting the appropriate options. The templates are saved in the directory specified by the environment variable `FEKO_USER_HOME`. If a new graph is created with GraphFEKO and the appropriate template file is not available, GraphFEKO will use internal defaults for the graph.

5.3.1.8 Load data

Use this command to plot data from a text file. It can be added to the current graph or a new graph can be created. The data must be in columns separated by one or more spaces (no commas). The data in the second and higher columns are plotted against the first column. The table below shows an example of a text file, with data for two lines, ready for import into GraphFEKO. The first line consists of the x axis caption (associated with the first column) followed by a legend for each of the remaining data columns.

Note that the captions must be in the first row and each caption must be between quotation marks. If no caption row is present, GraphFEKO will still import the data (first column on the x-axis, subsequent columns on the y-axis).

If any column (except for the first column) does not have the same number of data entries as the first column, the empty data points must be represented by the text NAN.

```
"Frequency [MHz]"  "mag{S_11}"   "mag{S_21}"
1.00000E+02        -4.74462E-01  -2.83472E+01
2.00000E+02        -1.46055E+00   2.66604E+01
3.00000E+02        -7.46769E+00   2.06073E+01
4.00000E+02        -2.19462E+00   NAN
5.00000E+02        -6.12814E+00   NAN
```

After selecting *Load data* a *Data import options* window is opened. By default a new graph window will be created. The data can also be imported onto the current active graph window by selecting *On current*. (Ensure that the correct graph window is active before selecting *Load data*.) For a new graph the user can also select between polar or line graphs. (For polar graphs the first column specifies the angle in degrees.)

For Smith chart data, the first column must contain the phase (in degrees) and the next columns the amplitude.

5.3.1.9 Save data

Export the data associated with the active graph window to a text based data file using the *Save data* option. The data will be saved in the GraphFEKO column format described in the previous section. Only the visible lines will be exported and a maximum of ten lines (11 columns in total) can be exported.

5.3.1.10 Print

The *Print* option invokes the *GraphFEKO printing* preview window. The user can specify the margins and page orientation. Changing the *Detail* setting towards *More* reduces the font size of all text and thus increases the actual graph area. This is useful when printing to high resolution printers.

5.3.1.11 Print to file

Select the *Print to file* option to print the graph on the active window to file. Currently Windows bitmap (`*.bmp`), jpeg (`*.jpg`), gif (`*.gif`), Windows meta file (`*.wmf`), enhanced Windows meta file (`*.emf`) and postscript files are supported. When printing to a bitmap format, the file will have the same size — in pixels — as the current graph on screen.

For the postscript printing a postscript printer must be installed on your system. (The Adobe PostScript printer, which can be down loaded from the Adobe web site, is a good option. Note that this printer can be set to print encapsulated PostScript, but the files will still have the `*.ps` extension.) For each GraphFEKO session you will need to select this printer when you print the first PostScript file.

5.3.1.12 Exit

Select the *Exit* option to exit GraphFEKO. All graphs will be closed and a warning will be given if the changes (or edits) made to any of the graphs have not yet been saved. You also have the option to quite without saving any unsaved graphs — if you answer *No* to this question you will be prompted for each unsaved graph.

5.3.2 Import menu

Used to load FEKO output data into GraphFEKO.

5.3.2.1 Select file

Use this option to select a FEKO output file (`*.out`). This is done automatically when GraphFEKO is executed from within WinFEKO, or if GraphFEKO is executed with a FEKO output file as command line parameter, e.g.

```
graphfeko.exe example_01.out
```

After selecting a file, the filename with extension `*.tmg` appears in the information bar at the bottom left of the GraphFEKO window, e.g. `C:\FEKO\examples\example_02.tmg` when the file `example_02.out` was selected. GraphFEKO creates this temporary file by copying the `*.out` file to a `*.tmg` file in the same directory. All subsequent file manipulations are done using the `*.tmg` file to avoid corruption of the `*.out` file.

5.3.2.2 Clear

Select this option to clear all data associated with the selected FEKO output file from GraphFEKO memory. The selected file is also cleared, i.e. a new FEKO output file must be selected before any further data extraction can be performed with GraphFEKO. This is used when doing additional runs with the same file name.

5.3.2.3 Antenna parameters

When this item is selected, GraphFEKO loads the antenna parameter data (if it exists) from the selected output file and opens the *Antenna parameters* panel.

The various frequencies at which the FEKO solution has been obtained are displayed in the *Frequency* box. The *Source nr* field lists the voltage sources in the order they appear in the output file. If a number of voltage sources are added before requesting output which requires a matrix solution, the sources will each be given a number such that one may plot the input parameters for the separate sources. The user should remember that this is then the resulting input parameters in the active environment which is not the same as exciting the sources one by one. If additional sources are added after the first calculation, this represents a new environment and the original sources will be listed again.

Click to highlight a frequency and select a source for this frequency. The corresponding *Source data* is displayed. This information has been extracted directly from the FEKO output file. `P_s` gives the available source power and `P_m` the power lost due to mismatch.

Under *Scale* the user can select a linear or logarithmic scale (to use a logarithmic scale, the plot parameter must remain larger than 0), or he can plot the quantity in dB. If any of the quantities selected have negative values and either *Log* or *dB* scaling has been selected, then a warning will be given and the graph will be plotted on a linear scale. One may also elect to plot the input impedance on the Smith chart.

Some antenna parameters can be plotted as a function of frequency if more than one frequency solution is available. To enable the *Plot options* select a range of frequency values in the frequency block by clicking and dragging over the required range.

The *Zo (ohm)* field gives the system impedance which is used for S_{11} and VSWR calculations. In some cases (such as when calculating coupling) the source segment is loaded with the system impedance. FEKO calculates an input impedance Z'_{in} which includes the loading impedance. One must therefor subtract the loading impedance from the calculated value to get the actual input impedance. If the source segment has been loaded with the system impedance (in the `*.pre` file) the *Subt Z_load* option should be checked, if no loading has been applied this box should be unchecked.

Plotting the input impedance on a Smith Chart If *Smith chart* is selected under the *Scale* options the input impedance is plotted on a Smith chart. This represents the magnitude and phase of S_{11}, but the grid is labeled according to the normalised impedance. (The chart is normalised to the impedance specified by the *Zo (ohm)* field discussed above.)

The grid labels can be switched off by editing the *Right* axis on the *Axis* tab of the *Chart* tab of the advanced editor (see section 5.3.3.2). The grid density can be controlled by setting the *Desired increment* field on the *Scales* tab of the same axis. Both the real and imaginary circles uses this increment between 0 and 1 and the inverse of these values between 1 and infinity.

The points on the Smith chart can be numbered to allow an indication of which frequency is associated with each point. Select the *Series* tab of the advanced editor, then select the appropriate series and check *Visible* box on the *Marks* tab.

Plotting the antenna parameters The *Quantity(s)* field allows the following antenna parameter quantities.

- **S_11 — the reflection coefficient S_{11} in a Z_0 ohm system**
  S_{11} is calculated from

$$S_{11} = \frac{(Z_{in} - Z_0)}{(Z_{in} + Z_0)}$$

  with Z_0 the specified impedance of the feed line. If *Subt Z_load* is checked as discussed above $Z_{in} = Z'_{in} - Z_0$ else $Z_{in} = Z'_{in}$.

- **VSWR — the standing wave ratio**
  The VSWR is calculated from

$$\text{VSWR} = \frac{1 + |S_{11}|}{1 - |S_{11}|}$$

  with S_{11} as calculated above.

- **Eff — the antenna efficiency**
  This is the efficiency of the antenna as calculated by FEKO. For a zero loss, ideal antenna this is 100% at each frequency but for more realistic antennas with losses the efficiency is a function of frequency.

- **Z_in — the antenna input impedance**
  This option plots the input impedance Z_{in}. As before, if *Subt Z_load* is checked $Z_{in} = Z'_{in} - Z_0$ else $Z_{in} = Z'_{in}$ where Z'_{in} is the input impedance calculated by FEKO with impedance loading included.

- **Y_in — the antenna input admittance**
  Plot the input admittance. The loading treatment is the same as for *Z_in*.

- **P_rad — the radiated power of the antenna**
  The power radiated from the antenna is plotted as calculated by FEKO. It is dependent on the input power as well as the efficiency and mismatch of the antenna.

For S_{11}, Z_{in} and Y_{in} one may plot the real and imaginary parts as well as the magnitude and phase or any combination of these components. The other parameters are all positive real values and one may only plot the magnitude.

The *New graph* button plots the frequency dependent data for the quantities selected on a new graph. The *Add to graph* button sends data for the selected quantities to the active graph window. The *Cancel* button closes the *Antenna parameters* panel and activates the *Main graph settings* panel.

5.3.2.4 S-parameters (SP card)

When this item is selected, GraphFEKO loads the S-parameters as calculated by FEKO on request of the SP card (see section 9.2.39) from the selected output file and opens the *Antenna S-parameters* panel.

The various frequencies at which the FEKO solution has been obtained are displayed in the *Frequency* box. The *Card nr* field lists the SP cards for each frequency. Click to highlight a frequency, or select a range of frequencies by clicking and dragging the mouse. If a range of frequencies that does not all have the same number of SP cards is selected, the frequencies with a different number of cards than the first is automatically unselected.

If more than one SP card is used for each frequency, the user can also plot the result as a function of the card number. (In this case only one frequency may be selected.) The is useful to plot the S-parameters as a function of other parameters in the `*.pre` file, for example the value of a terminating resistance.

Under *Scale* the user can select a linear or logarithmic scale (to use a logarithmic scale, the plot parameter must remain larger than 0), or he can plot the quantity in dB. If any of the quantities selected have negative values and either *Log* or *dB* scaling has been selected, then a warning will be given and the graph will be plotted on a linear scale. One may also elect to plot the input impedance on the Smith chart. (See the comments on setting labels and the grid density of Smith charts in the previous section.)

5.3.2.5 Network parameters

When this item is selected, GraphFEKO loads the network parameter data (if it exists) from the selected FEKO output file and activates the *Network parameters* panel. Network parameters only exist if the output file contains currents on segments (written with the OS card) which were calculated for voltage sources.

The various frequencies at which the FEKO solution has been obtained are displayed in the *Frequency* box. The *Source nr* field lists the sources for each frequency in the order they appear in the output file. (If two voltage sources are active at the same time, they are listed separately.) The *Source nr* field only affects I_1 and S_{11} information.

The *Block nr* field specifies the solution blocks available at each frequency. It is incremented each time the currents is recalculated. If only one source is active at a time, it should be equal to the *Source nr* field. The *Cur block* field allows the user to select the nth OS card for the given frequency and solution block. The *Segments* field then shows the segments available in that current block. Port 2 is assumed to be on the selected segment.

The *Labels* list is included for future use and is not active at present.

Click to highlight a frequency, and select a source and port 2 segment for this frequency. The data for the parameter selected under *Quantities* is displayed.

Plotting the network parameters as a function of frequency The network parameter quantities can be plotted as a function of frequency if more than one frequency solution is available. The *Parameter data* option panel will change to the *Plot options* panel when a range of frequency values in the frequency block are selected. (Click and drag with the mouse to select multiple frequencies.)

Under *Scale* the user can select a linear or logarithmic scale (to use a logarithmic scale, the plot parameter must remain larger than 0), or he can plot the quantity in dB. If any of the quantities selected have negative values and either *Log* or *dB* scaling has been selected, then a warning will be given and the graph will be plotted on a linear scale.

The *Subtract Z_load* field gives the system impedance which is used for S_{11} and S_{21} calculations. If S_{21} calculations are required, both the source and port 2 segment should be loaded with the system impedance (with, for example, LZ cards). If this is the case the *Subtract Z_load* should be checked. If the ports are not loaded, it should be left unchecked, but then S_{21} plots are not possible.

The *Quantity(s)* field allows the following parameter quantities.

- **S_11 — the reflection coefficient S_{11} in a Z_0 ohm system**
  See section 5.3.2.3 for a description of the S_{11} calculations. S_{11} is calculated for the source selected under *Source nr* and it is assumed that only one source is active.

- **S_21 — the transmission coefficient S_{21} in a Z_0 ohm system**
  For S_{21} calculation in a two-antenna system with Z_0 feed lines, the feed segment of each antenna must have been loaded with Z_0 (see also section 5.3.2.3) — then the *Subtract Z load* field must be checked.

  S_{21} is calculated from
  $$S_{21} = 2Z_o \frac{I_2}{V_{in}}$$
  with V_{in} the voltage applied to the transmit antenna and I_2 is the current on the receive antenna (Z_0 loaded) segment as calculated by FEKO.

S_{21} is calculated for a transmit antenna whose feed segment is selected in the *Source nr* drop down listbox and a receive antenna attached to the segment selected in the *Segments* listbox under *Target selection*. More than one segment in the *Segments* listbox can be selected for plotting on the same graph. The absolute segment numbers are listed and the user must ensure that the correct (or required) segment number is selected by verifying the segment number with WinFEKO.

- **I1 — the current I_1 on the source**

  This allows plotting the current on the various sources as selected under *Source nr*.

- **I2 — the current I_2 on the port 2 segment**

  This allows plotting the current on the selected receive port. More than one segment in the *Segments* listbox can be selected for plotting on the same graph.

For each of these quantities the user may plot the real and imaginary parts, the magnitude and phase. However, in most cases one may only plot either the real and imaginary parts or the magnitude or the phase at a time. (It is possible to use the *Add to graph* button to plot, for example, the magnitude of a current on the same graph as the real part.)

The *New graph* button plots the frequency dependent data for the quantities selected on a new graph. The *Add to graph* button sends data for the selected quantities to the active graph window.

The *Cancel* button closes the *Network parameters* panel and activates the *Main graph settings* panel.

Plotting the currents on a number of segments It is also possible to plot the current on a number of segments by selecting a single frequency and the I_2 quantity. We will discuss this in more detail in section 5.3.2.7.

5.3.2.6 Receiving antenna (Rx)

This item must be selected to extract induced current data on segments with a plane wave incident field as excitation. The *Antenna reception* panel in GraphFEKO is activated.

The various frequencies at which the FEKO solution has been obtained are displayed in the *Frequency* box. The *Inc field block* item represents different incident angles for a plane wave incident field. The *Cur block* field allows the user to select the nth OS card for the given frequency and incident direction. The *Segments* field then shows the segments available in that current block.

The *Labels* list is included for future use and is not active at present.

Click and highlight a frequency, and select an incident angle and a single output segment. The current on the segment is displayed under parameter data.

Under *Scale* the user can select a linear or logarithmic scale (to use a logarithmic scale, the plot parameter must remain larger than 0), or he can plot the quantity in dB. If any of the quantities selected have negative values and either *Log* or *dB* scaling has been selected, then a warning will be given and the graph will be plotted on a linear scale.

As before, the *New graph* button plots the frequency (or incident field) dependent data for the quantities selected on a new graph, the *Add to graph* button sends data for the selected quantities to the active graph window, and the *Cancel* button closes the *Antenna reception* panel and activates the *Main graph settings* panel.

Induced current as a function of frequency The antenna reception data (induced current, I_2, in a segment) can be plotted as a function of frequency if more than one frequency solution are available. Select multiple frequencies by clicking and dragging with the mouse in the *Frequency* list.

Now select the *Scale* and what to plot (real and/or imaginary parts, or the magnitude or the phase) and plot the current on the selected segment. More than one segment can be selected for current plotting, but only one incident direction can be selected.

Induced current as a function of incident angle If the FEKO solution contains more than one plane wave incident angle at one or more frequencies, the *Inc field block* will contain a list of items when the associated frequency is selected. Each item represents a different incident angle — in the order they are written to the output file. The current I_2 induced in a segment can be plotted as a function of incident angle. To do this select a single frequency and multiple incident directions. In this case the current can be plotted for only one segment at a time. The user can, of course, plot the current on different segments one after the other using the *Add to graph* button.

5.3.2.7 Current extraction

This allows the extraction of segment currents irrespective of the type of excitation. It is exactly analogous to the previous section except that the *Inc field block* is replaced by the *Excitation block* which lists the different excitations — it is incremented each time a solution is done for a different set of sources. The *Labels* list is included for future use and is not active at present.

As before, the *New graph* button plots the frequency (or incident field) dependent data for the quantities selected on a new graph, the *Add to graph* button sends data for the selected quantities to the active graph window, and the *Cancel* button closes the *Current extraction* panel and activates the *Main graph settings* panel.

Plotting the induced current as a function of frequency The induced current, I_2, in the specified segment can be plotted as a function of frequency in the same way as described for receiving antenna parameters in the previous section. Only one excitation may be selected, but multiple segments may be simultaneously plotted against frequency.

Plotting the induced current on a number of segments If a single frequency and a single excitation have been selected and multiple segment currents are available for this combination (as written with the OS card), a list of segments can be selected in the *Segments* list. The *Independent parameter* field then becomes visible.

If the *segnr* is selected in this field, the currents are plotted in the order they appear in the output file. This order depends on the way in which the model is created and usually does not reflect an intuitive order. The user may therefore elect to plot the segment currents as a function of one of the Cartesian coordinates. (This allows one, for example, to plot the current on a dipole antenna as a function of the position along the dipole.)

5.3.2.8 Far fields

When this item is selected, GraphFEKO loads the far field data (if it exists) from the selected FEKO output file. The *Far fields* panel in GraphFEKO is activated.

The various frequencies at which the FEKO solution has been obtained are displayed in the *Frequency* box. The *Block no* field lists the FF cards in the order they appear in the output file. Blocks which only integrate the fields, without writing field values to the output file, are not listed.

Under *Scale* the user can select a linear or logarithmic scale (to use a logarithmic scale, the plot parameter must remain larger than 0), or he can plot the quantity in dB. If any of the quantities selected have negative values and either *Log* or *dB* scaling has been selected, then a warning will be given and the graph will be plotted on a linear scale. In addition, the user can specify if a *Polar* (amplitude versus angle) or *Line* graph (standard rectangular 2D plot of the quantity value as a function of the independent variable) should be created.

As before, the *New graph* button plots the frequency (or direction) dependent data for the quantities selected on a new graph, the *Add to graph* button sends data for the selected quantities to the active graph window, and the *Cancel* button closes the *Current extraction* panel and activates the *Main graph settings* panel.

Various components can be plotted for each of the three quantities.

- **Gain/Directivity**
  GraphFEKO can plot the gain or the directivity depending on which one was requested by the far field card. If either gain or directivity is selected, a number of components are available as shown in figure 5-1.
  - *Total:* The total gain independent of the polarisation.
  - *Vertical:* The vertical (or ϑ) component.
  - *Horizontal:* The horizontal (or φ) component.

- *LHC:* The left hand circularly polarised component. The polarisation vector rotates counter clockwise when viewed, at a fixed position, in the direction of propagation.

- *RHC:* The right hand circularly polarised component. The polarisation vector rotates clockwise when viewed, at a fixed position, in the direction of propagation.

- *Z (+45 deg):* When viewed in the direction of propagation, the $\vec{e}_\vartheta$ unit vector points downwards and $\vec{e}_\varphi$ to the left. The unit vector for Z-polarisation is then $\vec{e}_Z = (\vec{e}_\vartheta + \vec{e}_\varphi)/\sqrt{2}$ which lies along an axis rotated +45 degrees from horizontal (in a counter clockwise direction) — coinciding with the direction of the diagonal line of the Z.

- *S (-45 deg):* Here $\vec{e}_S = (\vec{e}_\vartheta - \vec{e}_\varphi)/\sqrt{2}$ which is rotated by -45 degrees from horizontal and lies in the direction approximated by the diagonal of the S.

- **Electric field**

  For this only the *Theta* (or $\vec{e}_\vartheta$) and *Phi* (or $\vec{e}_\varphi$) components are available.

- **Axial ratio**

  The axial ratio of the radiated far field as calculated by FEKO. A negative sign indicates left hand polarisation. The sign is ignored when plotting the axial ratio in dB. This option is not available when a plane wave incident field is used as excitation.

- **RCS is the radar cross-section**
  The radar cross-section as calculated by FEKO. This option is only available with a plane wave incident field as excitation and if the appropriate far field calculations have been requested (see FF card).

Plotting the far field data as a function of angle If a single frequency is selected, the user can select between ϑ and φ as independent variable. The selected quantity will be plotted as a function of this value over the full available range. The input box for this value changes colour and shows the range. A single value must then be selected for the second variable.

Plotting the far field data as a function of frequency Some far field quantities can be plotted as a function of frequency if more than one frequency solution is available. The *Independent variable* under *Plot options* will change to *Freq. range*, when a range of frequency values in the frequency block is selected. This can be done by clicking and dragging over the required frequency range.

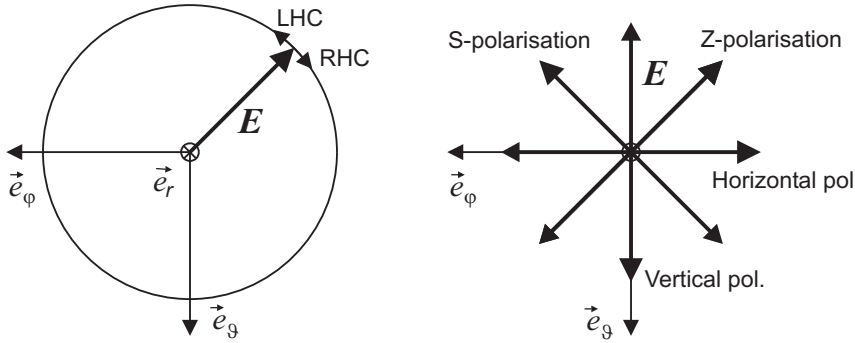


Figure 5-1: Polarisation of the electric far field as viewed, at a fixed position, in the direction of propagation.

5.3.2.9 Near fields

When this item is selected, GraphFEKO loads the near field data (if it exists) from the selected FEKO output file. The *Near fields* panel in GraphFEKO is activated.

The various frequencies at which the FEKO solution has been obtained are displayed in the *Frequency* box. The near field data blocks (one per FE card) available at that frequency are displayed in the *Block no* list.

Under *Scale* the user can select a linear or logarithmic scale (to use a logarithmic scale, the plot parameter must remain larger than 0), or he can plot the quantity in dB. If any of the quantities selected have negative values and either *Log* or *dB* scaling has been selected, then a warning will be given and the graph will be plotted on a linear scale. In addition, the user can specify if a *Polar* (amplitude versus angle) or *Line* graph (standard rectangular 2D plot of the quantity value as a function of the independent variable) should be created.

Plotting the near field data as a function of the coordinates If a single frequency is selected, it is possible to plot the near field as a function of any of the principle coordinates. Select a *Block no* and an *Independent variable* against which the data must be plotted (on the *Plot options* panel). The associated coordinate in the *Near field blocks* then changes colour. Select the correct value for the other two components in the lists on the *Near field blocks* panel.

In certain coordinate systems (or for separately specified points) some coordinates is dependent on each other). In this case more than one coordinate will change if the user change the value in one of the coordinate lists.

Plotting the near field data as a function of frequency Some near field quantities can be plotted as a function of frequency if more than one frequency solution is available.

Select a range of frequencies (click and drag with the mouse). The *Independent variable* under *Plot options* will change to *Freq. range*. Now select the coordinate values of the point for which the field must be plotted.

Selecting the plot options The desired *Near field quantity* and the *Component* of this quantity determine what field quantity to plot. The user may then select to plot the real and imaginary parts or the magnitude or the phase of the selected field quantity. Note that not all options are available for all quantities and not all quantities can be plotted simultaneously. (The user can, of course, add the additional quantity to the same plot by selecting this quantity and clicking the *Add to graph* button.) For example if *mag* is selected under *Components* (to give the total field at the specified position), only the *mag* option is available on the line below.

The quantities which can be plotted are:

- **E_field:**

  This is the electric near field as calculated by FEKO. This option is only available if the appropriate near field calculations have been requested (FE card).

- **H_field:**

  This is the magnetic near field as calculated by FEKO. This option is only available if the appropriate near field calculations have been requested (FE card).

- **Pointing vector** $S = \frac{1}{2}\,\mathrm{Re}[\vec{E} \times \vec{H}^*]$**:**

  This is the Pointing vector. The quantity is only available if *both* the electric and magnetic fields have been calculated with a single FE card in FEKO.

- **SAR:**

  This is the Specific Absorption Rate calculated in a lossy dielectric as:

$$SAR = \frac{1}{2}\frac{\sigma|\vec{E}|^2}{\rho}$$

  with σ the conductivity of the lossy dielectric, $|E|$ the magnitude of the electric field vector calculated by FEKO inside the dielectric and ρ the density of the dielectric material. Select the *More* button (top right of the *Near fields* panel) to change the ρ (density) value. (The conductivity is read from the `*.out` file.)

  This option is only available if the appropriate near electric field calculations (inside a dielectric) have been requested (see the FE card).

As before the *New graph* button writes the distance (or frequency) dependent data for the quantities selected to a new graph and the *Add to graph* button sends data for the selected quantities to the active graph window. The *Cancel* button closes the *Near fields* panel and activates the *Main graph settings* panel.

5.3.2.10 Adaptive frequency interpolation results (ADAPTFEKO)

This option is only available if ADAPTFEKO is enabled in the licence and a `*.afo` file is available for the currently selected results file. (The `*.afo` file is created by the FEKO solver if the FR card requests continuous data (see section 13). Note that one may select a `*.afo` file also without a `*.out` file.) It opens the *ADAPTFEKO results* panel from which one can plot the continuous frequency data. By its very nature it only allows plots as a function of frequency.

The *Parameter* field lists all the available quantities in a drop down list. The quantities are all that are also available in the `*.out` file. When the quantity is followed by `(sol=x,source=y)` it means that this quantity was calculated as part of the `x`th solution and the `y`th source. A new solution is considered each time the currents are recalculated. (The effect of frequency to cause a new solution is not considered here.)

Depending on the parameter the next blocks allow the user to select what he\she wants to plot. For impedance and admittance the *Subtract ...* option is similar to the case for the discrete frequency data. Note that subtracting 0.02 Siemens from the admittance is not the same as subtracting 50 Ω from the impedance. When plotting *S11* or *VSWR* one should use impedance for voltage type sources and admittance for current type source.

The *Number of increments* field indicates the number of points used in the graphical representation — the data is continuous. If the *Log scale* option after *Number of increments* is checked, the bottom axis is logarithmic and frequency points are spaced with a multiplicative constant rather than a linear spacing.

The modify button removes the last set of lines added to the graph and replaces them with the current setting. This is especially useful to change the number of frequency points, but one may also change the quantity or its components. This option is only available while the current graph stays unchanged. Once the user has shifted focus to a different graph, this option becomes unavailable.

The grey markers along the bottom axis indicate the discrete frequencies used by ADAPTFEKO. A high density of these markers indicates results that are not smooth. (Note that it is possible that there is a resonance in another parameter even if the current result seems quite smooth in such a band.) The frequency markers can be changed and/or hidden by clicking the *Bottom axis* button on the *Main graph settings* panel. Note that they are constructed when a graph is created from the *ADAPTFEKO results* panel. If continuous data is added to a previously existing graph, the frequency markers will be absent.

5.3.3 Edit menu

This menu item is used to select editing options for graphs created in GraphFEKO.

5.3.3.1 Edit graph

Select this item to activate the *Main graph settings* panel. The basic graph settings are available for editing on this panel. More advanced editing of a graph can be performed by clicking the *Advanced edit* button at the bottom right of the *Main graph settings* panel (see more detail on advanced editing in the following section).

The basic graph settings available for editing are:

- **Graph titles**
  Type in the required graph titles in the appropriate edit boxes available under *Graph titles*. The corresponding title on the active graph window will change while typing. Fonts and other graph title settings can be changed using advanced editing.

- **Graph settings**
  The *Auto-scale* option is selected by default. The minimum and maximum values for the active graph are displayed in the *Left axis min* and *Left axis max* edit boxes. Type a minimum or maximum value into these edit boxes to override auto-scaling. On changing the minimum or maximum values, the *Auto-scale* option is unselected.

  Normalisation can be off, each plot can be normalised to its maximum, or all plots can be normalised to the graph maximum. The *Norm type* drop down list gives the different normalisation options available. *Linear* normalises the data between 0 and 1; *Log* and *dB* between minus infinity and 0. *Note that the user has to select the correct normalisation type for the current graph.* Unselect the *Normalisation* checkbox to display the data without normalisation.

  The *Legend* option is selected by default. When selected, the legend associated with each line on the active graph is displayed. Unselect this option to remove the line legends from the active graph.

  The *Bottom axis* button (at the bottom of the panel) allows scaling and adding an offset to the horizontal axis of the graph.

- **Line settings**
  Select the line to edit from the *Line nr* drop down list. On selection, the basic line settings will be displayed under *Caption*, *Colour*, *Marker* and *Style*. Type the required legend caption for each line in the *Caption* edit box. The legend for the corresponding line will change on the active graph if the *Legend* option is selected under *Graph settings*.

  Choose a line colour, line marker and line style from the *Colour*, *Marker* and *Style* drop down lists. The corresponding line will change on the active graph. The *Visible* option is checked (selected) by default for each line. Unselect the *Visible* checkbox to hide (not delete) a specific line.

  The *Scaling* option allows the user to add a constant scaling factor to the data (on the vertical axis) and *offset* adds a constant offset to the data (such as required

for scaling dB graphs). This is useful, for example, to view small currents in mA rather than A. The scaling factor is not cumulative and each time it is changed the new factor is applied to the original data. When *Normalisation* is selected all plots are scaled before they are normalised. (Scaling only influence normalisation to the graph maximum, but if an offset has been added normalisation yields a different result.)

More line setting options are available under advanced editing.

- **Bottom axis**
  Click the *Bottom axis* button to set the scaling offset and limits of the bottom axis. *Auto* is selected by default, but this is unselected if the user changes the axis limits. Note that one must press <Enter> or exit the field after changing it before the changes will take effect.

  If the current graph was created with ADAPTFEKO results, it will contain markers showing the frequency samples along the bottom axis. If these markers are present they can be hidden by unchecking the *Show sample frequencies box*. One can also change their length and colour on this panel.

- **Zooming in on graph details**
  Zoom in on a specific region on a graph by clicking and dragging the mouse pointer on the graph (top left to bottom right). A rectangular window appears and when releasing the mouse button only the zoomed windowed region will be displayed on the graph. Unzoom by clicking the *Reset* button that appears in the top, left hand corner of the graph when zooming. This option is only available for line graphs.

- **Panning a graph**
  Pan a graph by right clicking with the mouse on the graph, hold down the right mouse button and move. "Unpan" by clicking the *Reset* button that appears in the top, left hand corner of the graph when zooming. This option is only available for line graphs.

5.3.3.2 Advanced edit

This menu item will activate the *GraphFEKO: Advanced editor*. Various graph properties can be set here under the *Chart* tab (top left). The *Series* tab (top right) can be selected to edit the line settings for the line series available on the active graph.

The details will not be discussed here. Most, if not all, of the options are self explanatory. Support on any of the settings can be obtained from

feko_support@emss.co.za

5.3.3.3 Copy graph

Copy the current graph to the Windows clipboard such that it can be pasted into another application. The graph is copied in the enhanced Windows metafile format.

5.3.4 Tools menu → Line arithmetics

The *Line arithmetics* item under the *Tools* menu allows simple arithmetics using two lines on the current graph. (If the current graph contains more lines, they must be deleted or made invisible — see the discussion of the *Main graph settings* in section 5.3.3.1.) The two lines must have the same x values — they are added on a point by point basis without checking the x axis values. The result is added as an additional line on the current graph.

5.3.4.1 Add two lines

This gives the sum of the two lines on the graph.

5.3.4.2 Subtract two lines

This has three sub items. Line A refers to the visible line with the lowest number and line B to the other line. One can plot the second line subtracted from the first, the first from the second, or the absolute value of the difference.

5.3.4.3 Multiply two lines

This gives the product of the two lines on the graph.

5.3.4.4 Re+Im -> Amp

This takes the magnitude of the two current lines, interpreted as the real and imaginary components, i.e. $Amp = \sqrt{Re^2 + Im^2}$.

5.3.4.5 Re+Im -> Phase

This determines the phase (in degress, all four quadrants) of the two current lines, interpreted as the real (line with the lowest number) and imaginary (line with the highest number) components, i.e. $Phase = \text{atan2}(Im, Re)$.

5.3.4.6 Amp+Phase(deg) -> Re

This calculates the real part, interpreting the two current lines as the magnitude (line with the lowest number) and phase (in degrees, line with the highest number), i.e. $Re = Amp * \cos(Phase)$.

5.3.4.7 Amp+Phase(deg) -> Im

This calculates the imaginary part, interpreting the two current lines as the magnitude (line with the lowest number) and phase (in degrees, line with the highest number), i.e. $Im = Amp * \sin(Phase)$.

5.3.4.8 Log_10(Abs(Line A))

This takes the logarithm to the base 10 of a graph. It can be useful to convert values to dB if they were not plotted as such. The absolute value is taken before taking the log. This item can be called with one or two lines visible on the graph — the operation is performed on the visible line with the lowest number.

5.3.4.9 Sqrt(Abs(Line A))

This allows taking the square root of a graph. The absolute value is taken to avoid imaginary numbers. This item can be called with one or two lines visible on the graph — the operation is performed on the visible line with the lowest number.

5.3.4.10 Sqr(Line A)

This allows taking the square of a graph. This item can be called with one or two lines visible on the graph — the operation is performed on the visible line with the lowest number.

5.3.5 Tools menu → Unwrap phase

The *Unwrap phase* item is toggled between checked and unchecked. When this item is checked, the plots on the graph are considered to be phase plots and are unwrapped where they jump through 360 degrees.

5.3.6 Window menu

The Window menu controls the display and arrangement of the multiple graph windows in the main GraphFEKO display.

5.3.6.1 Cascade

This option arranges the graph windows stacked behind each other with the top name bars of all graph window visible.

5.3.6.2 Tile

This option arranges the graph windows in horizontal tiles.

5.3.6.3 Arrange icons

This option only applies to the icons of minimised graphs. It arranges the icons associated with all minimised graph windows next to each other at the bottom left of the main GraphFEKO display.

5.3.6.4 Minimise all

This option minimises all graph windows.

5.3.6.5 List of windows

You can select a graph window by clicking on the graph window or by selecting the name of the graph file (associated with a graph window) from the graph files listed under the Windows menu option.

5.3.7 Help menu

5.3.7.1 User's manual

Select *User's manual* item under *Help* to load the FEKO *User's Manual* in the PDF viewer associated with `*.pdf` files on your system. If nothing happens when this item is selected, then an Acrobat Reader has probably not been installed on your system. The Acrobat Reader can be installed from the FEKO CD.

5.3.7.2 About GraphFEKO

Select this item to get the GraphFEKO version number and development information.

6 The preprocessor PREFEKO

6.1 Description

The surface of the structure to be analysed with the program FEKO, has to be subdivided into elementary surfaces (in this case triangles). Wires have to be subdivided into segments. The mesh size is dependent on the wavelength in the medium surrounding the structure. The program PREFEKO can do all the meshing. It automatically generates the geometric data, in the form required by FEKO, from the data given by the user. The mesh density is controlled by a couple of parameters.

This section describe the principle workings of the program. The user first defines the location of points in space with the DP card. Structures are then defined in terms of these points. For example, two points may be joined to form a line (BL card), or four points for a parallelogram (BP card).

6.2 Running PREFEKO

If, for example, a file `example.pre` has been created using a text editor, PREFEKO is started using the following command:

```
prefeko example
```

After successful execution a file `example.fek`, is created. This is the input file for FEKO. The program PREFEKO allows a number of options, which are mainly used for debugging purposes. Entering `prefeko` without arguments will give an overview of the syntax and supported options. If, for example, the argument `--fek-format x` is specified, PREFEKO creates a `*.fek` file using the xth file format.

6.3 Symbolic variables

Instead of using numerical values in all the different cards, it is possible to use predefined variables. The name of a variable always consists of a `#`-sign followed by a string consisting of the characters `a-z`, `A-Z`, `0-9` and the special character `_`. The following are valid variable names, `#height`, `#a`, `#STARTINGFREQUENCY`, `#a_1` or `#P5_7f`, while the following are not valid: `#a?1` or `#value2.1`. There is no distinction between upper and lower case characters. For example, `#a` and `#A` is interpreted as the same variable.

Expressions and functions may be used when defining variables, so that direct calculations can be carried out. The variables have to be defined before they can be used in the respective cards. It is possible to use expressions like `2*#radius` in the input fields subject to the maximum allowed length (10 characters for real values, 5 characters for integer values). For larger expressions additional variables have to be defined.

A variable is defined in the following way:

```
#pi = 3.1415
#2pi = 2*#pi
#vara = 1 + sqrt(2)
#varb = #vara * 2.3e-2 * (sin(#pi/6) + sin(rad(40)) + #vara^2)
#summe = #vara+#varb
```

Note that the # sign has to appear in the first column. On the right hand side of any expression, variables that have already been defined can be used in conjunction with any of the following functions:

()	brackets
+	addition
−	subtraction
*	multiplication
/	division
^	powers, for example 2^3=8
SIN	sine (argument in radians)
COS	cosine (argument in radians)
TAN	tangent (argument in radians)
COT	cotangent (argument in radians)
ARCSIN	arcsine (in radians)
ARCCOS	arccosine (in radians)
ARCTAN	arctangent (in radians)
ATAN2	this function has two arguments `atan2(#y,#x)` it yields `arctan(#y/#x)` in the range $-\pi \ldots \pi$
ARCCOT	arccotangent
SINH	hyperbolic sine
COSH	hyperbolic cosine
TANH	hyperbolic tangent
SQRT	square root
LOG	logarithm to the base 10
LN	natural logarithm
EXP	exponential function
ABS	absolute value
DEG	convert radians into degrees
RAD	convert degrees into radians
STEP	step function, i.e. $\text{STEP}(x) = 0$ for $x \leq 0$; $\text{STEP}(x) = 1$ for $x > 0$
CEIL	smallest integer value that is equal to or greater than the argument
FLOOR	largest integer value that is equal to or smaller than the argument

MAX	returns the largest of the two arguments — called as `max(#a,#b)`
MIN	returns the smallest of the two arguments — called as `min(#a,#b)`
FMOD	this function also has two arguments `fmod(#a,#b)` and returns the remainder of the division `#a/#b`
RANDOM	This function returns a random value in the range 0 .. 1. If the argument X of RANDOM() is -1, then a random number is returned. For any other argument X in the range 0 .. 1 this value is used to set the seed, and then a random number is created using this seed. (Using the same seed allows one to create a deterministic and reproducible random number series). If "RANDOM(-1)" is called before any seed is set in the `*.pre` file, then the returned values are random and not reproducible. (The internal seed is used based on the time when PREFEKO is executed).

X_COORD	This function returns the x coordinate of a point previously defined by a DP card. The name of the point, in quotation marks, is passed as an argument to the function, for example
	`DP PNT01 1.234 0.4567 #z`
	`#x = x_coord("PNT01")`
	will set the parameter `#x` equal to 1.234

Y_COORD	Returns the y coordinate of a previously defined point similar to the function X_COORD discussed above.
Z_COORD	Returns the z coordinate of a previously defined point similar to the function X_COORD discussed above.

The FILEREAD function reads a numerical value from an arbitrary ASCII file. The general syntax is

```
fileread("Filename", Line, Column)
```

and contains the filename, the line number to read from and the column to read. (The data in the respective columns of any line are separated by one or more spaces.) For example with a file containing

```
Frequency in MHz      Re(load) in Ohm    Im(load in Ohm)
   100                    22.54              -12.56
   150                    25.07               -6.54
   200                    27.42                0.23
```

the frequency and loading can be imported directly from the file

```
#numfreq = 3    ** Number of frequencies
!!for #i = 1 to #numfreq
```

```
** Define the frequency (conversion from MHz to Hz)
#freq = 1.0e6*fileread("datafile.dat", #i+1, 1)
FR   1    0                    #freq

** Define the load
#Zr = fileread("datafile.dat", #i+1, 2)
#Zi = fileread("datafile.dat", #i+1, 3)
LZ   0                         #Zr       #Zi

** Computations ...

!!next    ** End of frequency loop
```

In addition to these functions PREFEKO allows the use of logical operations. It supports the function NOT() — which returns TRUE if the argument is FALSE and FALSE when the argument is TRUE — and the delimiters `>`, `<`, `>=`, `<=`, `=`, `<>`, AND and OR. When boolean operations are applied to variables, a value of 0 is taken as FALSE and everything else is interpreted as TRUE. Similarly in the result of a logical operation FALSE is mapped to 0 and TRUE to 1.

PREFEKO also supports a logical function DEFINED(`#variable`) which returns TRUE if a the variable `#variable` has been defined, and FALSE if not. This is useful in `*.pre` files where during OPTFEKO, TIMEFEKO or ADAPTFEKO runs certain variables are inserted into the header of the file. One can now process and view such `*.pre` files. For example in a `*.pre` file which will be optimised with respect to the variable `#a`, one may use

```
!!if (not(defined(#a))) then
#a = 200.0e-3
!!endif
```

The order of precedence is (lower levels are evaluated first) is

OR
AND
`=` and `<>`
`>`, `<`, `>=` and `<=`
`+` and `-`
`*` and `/`
`^`
`+` and `-` (when used as sign)
function calls
single number, expressions in brackets

Some variables are predefined in PREFEKO, but may be overwritten by re-assignments. These are:

Name	Value	Description
`#pi`	$3.14159265358979\ldots$	The constant π
`#eps0`	$8.85418781761 \cdot 10^{-12}$	Dielectric constant ε_0 of free space
`#mu0`	$4\,\pi \cdot 10^{-7}$	Permeability μ_0 of free space
`#c0`	$\frac{1}{\sqrt{\mu_0\,\varepsilon_0}}$	The speed of light in free space
`#zf0`	$\sqrt{\frac{\mu_0}{\varepsilon_0}}$	The intrinsic impedance of free space
`#true`	1	Used for logical true
`#false`	0	Used for logical false

There are three other special variables `#!x`, `#!y` and `#!z` which are very useful for the connection of complex wire structures. The three variables specify the Cartesian coordinates of the end point of the wire segment most recently defined. This enables the correct and easy connection of a straight wire to a curved length of wire, as the next extract from an input file demonstrates:

```
CL        .....
DP        A                    #!x      #!y       #!z
#z = #!z + 0.5
DP        B                    #!x      #!y       #z
BL        A    B
```

The following example demonstrates the use of variables.

```
** A dielectric sphere in the field of an incident wave

** Define the variables
#r = 1          **   Radius of the sphere
#betrad = 1     **   Electrical size of the sphere
#epsr = 15      **   The relative dielectric constant
#maxlen = 0.7   **   The maximum edge length

** Define segmentation parameters
IP                                      #maxlen

** The corner points
DP     A                       0        0         0
DP     B                       0        0         #r
DP     C                       #r       0         0

** Select the medium
ME     1    0
```

```
** Generate an eighth of the sphere
KU     A    B    C             0         0         90        90        #maxlen

** Use symmetry in all three coordinate planes
**   yz-plane: ideal electrically conducting plane
**   xz-plane: ideal magnetically conducting plane
**   xy-plane: only geometrically symmetric
SY   1    2    3    1

** End of the geometry
EG   1    0    0    0    0

** Assigning the dielectric's properties
DI                           #epsr     1.0

** Incident plane wave excitation
#freq = #betrad * #c0 / (2*#pi*#r)
FR   1    0                  #freq
A0   0         1    1          1.0       0.0      -180.0

** Near fields along the z axis
FE   1    1    1    25   0     0.0       0.0      -1.98     0.0      0.0      0.04
FE   4    1    1    50   0     0.0       0.0      -0.98     0.0      0.0      0.04
FE   1    1    1    25   0     0.0       0.0       1.02     0.0      0.0      0.04

** End
EN
```

The use of variables makes the investigation of structures with varying geometry (e.g. variable distance of the antenna in front of a reflector) an easy process, because only one variable needs to be changed. It also allows FOR loops and IF conditions.

6.4 FOR/NEXT loops

Some cards in FEKO implicitly use loops (such as when an FR card with multiple frequencies is used). This, however, does not always offer the flexibility which one may require, for example, to change the material parameters inside the loop. Another example would be the use of a loop to create a complex geometry.

For completely general loops, PREFEKO allows the construct

```
!!for #var = #start to #end step #delta

!!next
```

where a simple example would be

```
** Loop for the relative permittivity
!!for #epsr_r = 1 to 5 step 0.5

** Set material parameters
GF   0                        #eps_r    1

** Compute fields etc.
FE

** End of loop
!!next
```

The syntax requirements of FOR/NEXT loops are:

- The `!!` characters must be located in the first two columns of the line. This is followed by a number of optional spaces and the keyword `for` (it is not case sensitive, so also `FOR` or `For` are accepted).
- The keyword `for` is followed by the name of the loop variable (starting with `#`). The variable name is terminated by a space or the `=` sign.
- Next follows an expression for the initial value of the loop (a constant, variable or formula, see the example below).
- This is followed by the keyword `to` and the terminating value of the loop variable (again a constant, variable or formula).
- The default increment of the loop variable is 1, but it can be changed by using the keyword `step` followed from an expression. Negative increments are allowed.
- The loop is terminated by a line of the form `!!next` (spaces are allowed between `!!` and `next` but not before the `!!`). All instructions and input cards between `!!for` and `!!next` are evaluated repeatedly inside the loop.
- Several loops can be nested as shown in the example below.

A more complicated example, illustrating some of the points above, is as follows

```
#end = 3+sin(4)
!!for #x1 = sqrt(5) + 2*3 to 2*#end step -#end/10
!!  for #x2 = 1.23 to 2*#x1   ** this is the inner loop
```

```
#x3 = #x1 + #x2
DP ....
.... (more commands)

!!  next
!!next
```

6.5 IF/ELSE/ENDIF constructs

This construct is used to allow different control cards under different conditions. The syntax requirements of IF/ELSE/ENDIF constructs are:

- The `!!` characters must be located in the first two columns of the line. This is followed by an arbitrary number of spaces, the expression to evaluate evaluated and the keyword `then` (it is not case sensitive, `THEN` or `Then` are also accepted).
- The block is terminated by a line of the form `!!endif` (again spaces are allowed between `!!` and `endif` but not before the `!!`).
- An optional line of the form `!!else` (again, the `!!` must be in the first two columns and spaces are allowed before the keyword which is not case sensitive).
- All instructions and input cards between `!!if` and `!!endif` (or `!!else` if it is present) are processed if the expression is TRUE. If it is present, all lines between `!!else` and `!!endif` are processed if the expression is FALSE.

Examples, illustrating some of the points above, are as follows

```
!!if #a > 5 then

...

!!endif
```

or

```
#l = (#a+5 > 21) and (#a < 100)
!!if ( (3*#a+5 >= #x/2) and not(#l) ) then

...

!!else
```

```
!!  if (sin(#x/10) > 0.5) then

...

!!  else

...

!!  endif
!!endif
```

6.6 Symbolic node names

When defining or using node names, simple variable names of the form `A#i` are allowed. The algorithm is that if a hash sign is found in a node point name, this hash sign and everything that follows is interpreted as a variable string, evaluated and rounded to the nearest integer. Thus if we have `#k=15` and use or define a point `P#k` then this is equivalent to using `P15` as point name. The length of the node name string (before and after expansion) is still limited to 5 characters.

For instance, it would now be possible to define the points P1 to P20 inside a loop.

```
!!for #k = 1 to 20
DP   P#k           .....
!!next
```

6.7 PRINT and EXIT commands

PREFEKO also supports the command `!!exit` to stop execution and the `!!print` command to print strings (enclosed in double quotes) and floating point numbers. The `!!print` command command accepts multiple arguments separated by commas. For example,

```
!!print "2*#b = ", 2*#b
!!if #a < 2*#b then
!!  print "The value of #a is too small:", #a, " (exiting now)"
!!  exit
!!endif
```

will print a warning and exit if the variable `#a` is smaller than two times variable `#b`.

6.8 Copyright to Voronoi

The preprocessor PREFEKO uses part of the program `voronoi` to execute a Delaunay triangulation, for some of the geometric cards. The parts of this program may be used freely and the required Copyright declaration is supplied in the accompanying documentation. The Copyright declaration follows:

```
/*
 * The author of this software is Steven Fortune.  Copyright (c) 1994 by
 * AT&T Bell Laboratories.
 * Permission to use, copy, modify, and distribute this software for any
 * purpose without fee is hereby granted, provided that this entire notice
 * is included in all copies of any software which is or includes a copy
 * or modification of this software and in all copies of the supporting
 * documentation for such software.
 * THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED
 * WARRANTY.  IN PARTICULAR, NEITHER THE AUTHORS NOR AT&T MAKE ANY
 * REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY
 * OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.
 */
```

7 The program FEKO

7.1 Introduction

The program FEKO does the actual field calculation. Input and output are done using files. The program indicates, on screen, how far the calculation has progressed.

7.2 Running FEKO

7.2.1 Running the sequential version

When FEKO is not executed from EditFEKO or WinFEKO, it can be started in a DOS window (on PC) or a shell (in UNIX) by executing the command

```
runfeko example_08
```

where `example_08.pre` must be an existing input file. RUNFEKO executes PREFEKO if the `*.fek` file is missing or older than the `*.pre` file and then executes the appropriate FEKO solver (sequential, parallel or adaptive sampling). It accepts the following optional parameters (see also section 7.2.2 for additional options to launch and control the parallel version of the solver)

`--priority x`	The value `x` specifies the CPU usage priority of the FEKO run: 0 = idle, 1 = below normal, 2 = normal, 3 = above normal and 4 = high. If not specified, the default is 2. This option might not be available for specific systems or specific FEKO versions, then it is just ignored.
`--prefeko-options ...`	All options following this (if one is used, up to the next `--xxx-options`) are passed to PREFEKO.
`--feko-options ...`	All options following this (if one is used, up to the next `--xxx-options`) are passed to FEKO.
`--adaptfeko-options ...`	All options following this (if one is used, up to the next `--xxx-options`) are passed to ADAPTFEKO.

The optional command line parameters for FEKO (specified after `--feko-options`) are

`--check-only`	If this option is used, FEKO processes and checks the geometry, but does not start a solution. This is useful to test an input file on a local PC before submitting it to a parallel computer.
`-e ENV=value`	This has the same effect as starting FEKO with the environment variable `ENV` set to `value`. More than one `-e ...` argument is allowed.

7.2.2 Running the parallel version

The parallel version is started with (on Windows it can also be configured and started from WinFEKO, see section 3.5.4)

```
runfeko example_08 -np x
```

where the parameter `x` following `-np` gives the required number of processes. In addition to the arguments listed in section 7.2.1, the parallel version accepts the following optional parameters

`-np x`	Start the parallel FEKO version with x processes.
`--machines-file machname`	The file `machname` is the machines file with the node names and the number of CPUs (see below).
`--mpi-options ...`	All options following this (if another `--xxx-options` parameters is used, all arguments before the second `--xxx-options` parameter) are passed to the MPI launcher (e.g. mpirun).

The number of processes to start on each available host is specified in a so-called *machines file* with the general syntax

```
Hostname:Number of processes
```

using a new line for each host. For example, if the user has two hosts with names `host1` and `host2` (this is the output of the UNIX command `hostname`), and 4 and 8 processors respectively, the machines file will be

```
host1:4
host2:8
```

With this machines file, the example with 6 processes given above will run with 4 processes on `host1` and 2 on `host2`.

If on one host only one process shall be started, then instead of the entry `host3:1` in the machines file, also the shorter form `host3` can be used.

Such a machine file (the file `mpi/share/machines.feko` under the FEKO installation path `FEKO_HOME`) is automatically created during the installation of the parallel version of FEKO. By default FEKO uses this file. If a different distribution of the processes is required, one can edit this file manually. This is, however, strongly discouraged. The user should rather create a separate machines file with the syntax described above. If this file is, for example `machname`, the environment variable `FEKO_MACHFILE` is used to force RUNFEKO to use this file instead of the default. The required commands (for the sh shell) are

```
FEKO_MACHFILE=./machname
export FEKO_MACHFILE
runfeko example_08 -np 6
```

Alternatively one may pass the name of the machines file as an argument to RUNFEKO.

```
runfeko example_08 -np 6 --machines-file ../../mymachines
```

Using RUNFEKO is independent of the respective platforms and MPI implementations (see also the discussion of the environment variable `FEKO_WHICH_MPI` in section 2.7). For very special applications or experienced users it may be necessary to pass additional options to MPI. (In such a case, the appropriate MPI manuals — located in the subdirectory `$FEKO_HOME/mpi/doc` — should be considered.) These are added after the argument `--mpi-options`. For example on the ScaMPI cluster (assuming `FEKO_WHICH_MPI=6`), the call

```
runfeko example_08 -np 6 --mpi-options -immediate_handling \
   threaded -smtrace 5-6
```

(all on one line) is interpreted internally and FEKO is executed with the command

```
/opt/scali/bin/mpimon -export env -immediate_handling threaded \
   -smtrace 5-6 /opt/feko/bin/feko.csv example_08 -- host1 4 \
   host2 2
```

(Note that `host1` and `host2` are examples only — the actual information is taken from the machines file.)

8 Description of the geometry cards

8.1 Overview of the geometry cards

The following table lists all input cards that are used to create the geometry (i.e. the cards that appear before the EG card in the `*.pre` file[5]). Most of these cards are processed by PREFEKO. For example, PREFEKO processes the BP card and writes the triangle elements to the `*.fek` file as input to FEKO.

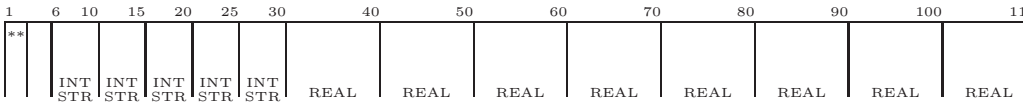
Card	Description
`**`	characters used to indicate a comment
`BL`	creates a line
`BP`	creates a parallelogram
`BQ`	creates a quadrangle
`BT`	creates a triangle
`CB`	changes already assigned labels
`CL`	creates a circular line using segments
`CN`	change the direction of the normal vector
`DK`	dielectric or magnetic eighth of a sphere
`DP`	define a node point
`DZ`	dielectric cylindrical shell
`EG`	end of the geometric input
`EL`	generation of a segment of an ellipsoid
`FO`	defines a Fock area
`HE`	creates a coil from wire segments
`IN`	reads an external *Include* file containing the geometric information
`IP`	sets the parameter that defines the degree of meshing
`KA`	defines the border of the PO area
`KK`	creates a circular conical segment
`KL`	sets the wedges in the PO area
`KR`	creates a circular element
`KU`	creates a spherical element
`LA`	specifies the label for segments, triangles, polygons, etc.
`ME`	defines the medium
`NU`	specify a NURBS surface from specified control points
`PB`	generation of a paraboloid
`PH`	create a flat plate with an elliptic hole
`PM`	create a polygonal shape that is meshed into triangles
`PO`	applies the Physical Optics approximation

[5] In general all the geometry cards must appear before the EG card. Exceptions are the IN card when including `*.pre` files with control commands; and the DP and TP cards which may be used to define points for the AP card.

PY	creates a polygonal surface for use with UTD or PO
QU	creates a dielectric or magnetic cuboid
SF	enters a scaling factor, with which all dimensions are multiplied
SU	switches the program into the superuser mode
SY	utilise symmetry in the construction of the geometry
TG	transformation (i.e. translation and rotation) of the geometric structures
TO	creates a toroid
TP	transform a point
UT	parameters for the uniform theory of diffraction (UTD)
UZ	creates a cylinder for use in the UTD region
WG	creates a parallelogram consisting of a wire grid
ZY	creates a cylindrical element

8.2 Alphabetical description of the geometry cards

8.2.1 ** Card



The ** card is not a command, but defines a comment line. Everything that is found in this line is ignored by PREFEKO.

Normally the filename has to be found in the first line of the (`*.pre`) file. If the first line happens to be comment line, then the filename will automatically be placed in the first line of the file `*.fek`, by PREFEKO.

It is possible to add a comment to the end of an existing line or card. For example,

```
** Definition of Parameters
#lambda = 1.0          ** Wavelength
#radius = #lambda/2    ** Cylinder radius
#height = 2*#lambda    ** Cylinder height
```

8.2.2 BL Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
BL		S_1	S_2				R_1	R_2						
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

Using this card two points are connected to form a line, which is then subdivided into segments. The points have to be defined by a DP card, prior to using this card. The wire radius is set by an IP card preceding the BL card, but can be set locally. Figure 8-1 shows an example.



Figure 8-1: Sketch illustrating the use of the BL card

Parameters:

- S_1 The name of the begin point of the line.
- S_2 The name of the end point of the line.
- R_1 Normally the wire radius is set with the IP card. Setting R_1 overrides this radius for the current wire without affecting the default for later segments. (R_1 is in m and is affected by the SF card scaling factor.)
- R_2 By setting the parameter R_2 it is possible to create a wire with a tapered radius. The parameter R_1 must also be set, R_1 then specifies the radius at point S_1 while R_2 specifies the radius at S_2.

Examples of BL card usage:

The following commands create the segmented wire shown in figure 8-2.

```
**
IP                              0.02                0.15
DP   A                          0.0         0.0     0.0
DP   B                          0.0         0.0     1.0
BL   A    B
EG
EN
```



Figure 8-2: Example of a BL card

These commands create the tapered wire shown in figure 8-3.

```
**
IP                                                  0.15
DP   A                          0           0       0
DP   B                          1           0       0
#rad = 0.005
BL   A    B                     #rad        5*#rad
EG
EN
```



Figure 8-3: Example of a BL card with a tapered radius

8.2.3 BP Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
BP		S_1	S_2	S_3	S_4		R_1	R_2						
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card four points are connected to each other to create a parallelogram. This parallelogram will then be subdivided into triangular elements. Figure 8-4 shows a sketch.

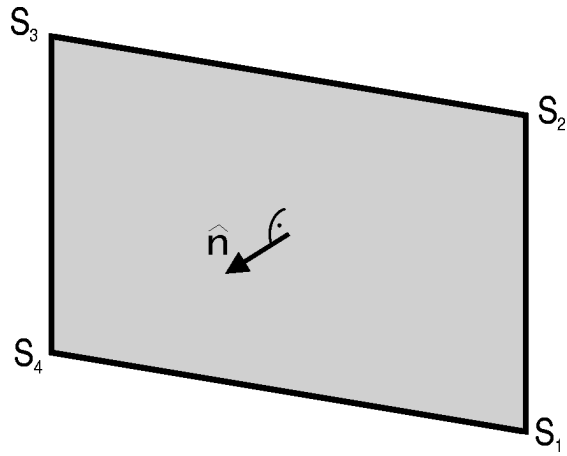


Figure 8-4: Sketch illustrating the use of the BP card

Parameters:

- S_1 Name of the first point of the parallelogram.
- S_2 Name of the second point of the parallelogram.
- S_3 Name of the third point of the parallelogram.
- S_4 Name of the fourth point of the parallelogram.
- R_1 Normally a parallelogram is segmented according to the edge length specified with the IP card. In some cases, e.g. when creating small microstrip lines, it may be desirable to use a finer segmentation in one direction. If set, R_1 specifies the maximum edge length of the triangle elements along the edges S_1–S_2 and S_3–S_4 (they have the same length).
- R_1 Similar to R_1, but applies to the edges S_2–S_3 and S_4–S_1.

The points that have been previously defined in the DP card are connected in the order in which they appear in the BP card. Thus the user has to ensure that the points describe

a parallelogram. If this is not the case, then PREFEKO will abort with the appropriate error message. The BQ card is used to generate quadrangles.

The direction of the normal vector ($\hat{n}$) of the subdivided triangles is determined by the right hand rule, through all the corners. This direction only has meaning when used with the Physical Optics (PO card, section 8.2.29) or with dielectrics (ME card, section 8.2.24).

Example of BP card usage:

Through the use of the following commands the parallelogram in figure 8-5 is created.

```
**
IP                                          0.2
DP   A                          0.0         0.0         0.0
DP   B                          0.0         0.0         1.0
DP   C                          1.0         0.0         1.0
DP   D                          1.0         0.0         0.0
BP   A    B    C    D
EG
EN
```

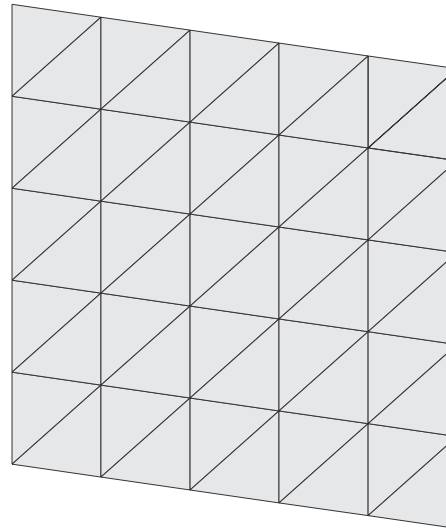


Figure 8-5: Example for the BP card

8.2.4 BQ Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
BQ	S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4					
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

With this card four points are connected to form a quadrangle. The quadrangle is then subdivided into triangles. An example is shown in figure 8-6.

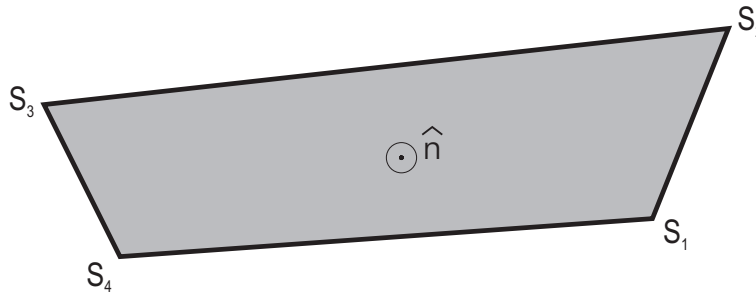


Figure 8-6: Sketch illustrating the use of the BQ card

Parameters:

- S_1 Name of the first point of the quadrangle.
- S_2 Name of the second point of the quadrangle.
- S_3 Name of the third point of the quadrangle.
- S_4 Name of the fourth point of the quadrangle.
- R_1 Normally the quadrangle is segmented according to the triangle edge length specified with the IP card. However, it is often desirable to have an inhomogeneous segmentation, for example in the transition from a finely segmented region to a region with coarser segmentation. If set, the parameter R_1 specifies the triangle edge length along the edge S_1–S_2. (R_1 is in m and is affected by the SF card scaling factor.)
- R_2 Similar to R_1, but applies to the edge S_2–S_3.
- R_3 Similar to R_1, but applies to the edge S_3–S_4.
- R_4 Similar to R_1, but applies to the edge S_4–S_1.

The points have to be predefined using DP cards prior to the BQ card and are connected in the order in which they appear in the BQ card.

In principal the BQ card can create all types of quadrangles, including parallelograms. The difference is that the BP card creates a regular subdivision.

The direction of the normal vector ($\hat{n}$) of the subdivided triangles is determined by the right hand rule through all the corners. This direction only has meaning when used with the Physical Optics (PO card) or with dielectrics (ME card).

Example of BQ card usage:

The commands below will create the quadrangle in figure 8-7.

```
**
IP                                        0.3
DP   A                          0.0       0.0       0.0
DP   B                          0.0       0.0       1.0
DP   C                          1.0       0.0       1.3
DP   D                          0.8       0.0       0.1
BQ   A    B    C    D
EG
EN
```

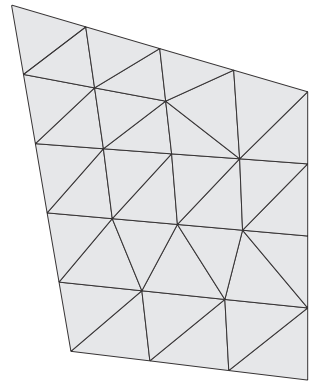


Figure 8-7: Example of a BQ card

A plate with a finely segmented slot, as shown in figure 8-8, can be created with the following sequence of commands.

```
** Nonuniform meshing for a slot in an aperture
** all dimensions in mm
SF   1                          0.001
#len = 20           ** length of the aperture
#wid = 2            ** width of the aperture
#pl_len = 50        ** length of (nonuniform mesh) plate around slot
#pl_wid = 30        ** width of (nonuniform mesh) plate around slot
#tot_len = 120      ** total length of the plate
#tot_wid = 60       ** total width of the plate

#edge_ap   = 1.5    ** edge length around the aperture
#edge_glob = 5      ** global edge length
IP                                        #edge_glob
```

```
** define some points
DP   A                          0          0          #wid/2
DP   B                          #len/2     0          #wid/2
DP   C                          #len/2     0          0
DP   D                          #pl_len/2  0          0
DP   E                          #pl_len/2  0          #pl_wid/2
DP   F                          0          0          #pl_wid/2
DP   G                          #tot_len/20           0
DP   H                          #tot_len/20           #pl_wid/2
DP   I                          #tot_len/20           #tot_wid/2
DP   J                          #pl_len/2  0          #tot_wid/2
DP   K                          0          0          #tot_wid/2

** create the geometry
#edge_x = 1.0*(#edge_glob+#edge_ap)/2   ** for diagonals
BQ   A    B    E    F          #edge_ap  #edge_x              #edge_x
BQ   B    C    D    E          #edge_ap  #edge_x              #edge_x
BP   E    D    G    H
BP   J    E    H    I
BP   K    F    E    J

** create the full plate with symmetry
SY   1    1    0    1

** end of geometry
EG
EN
```

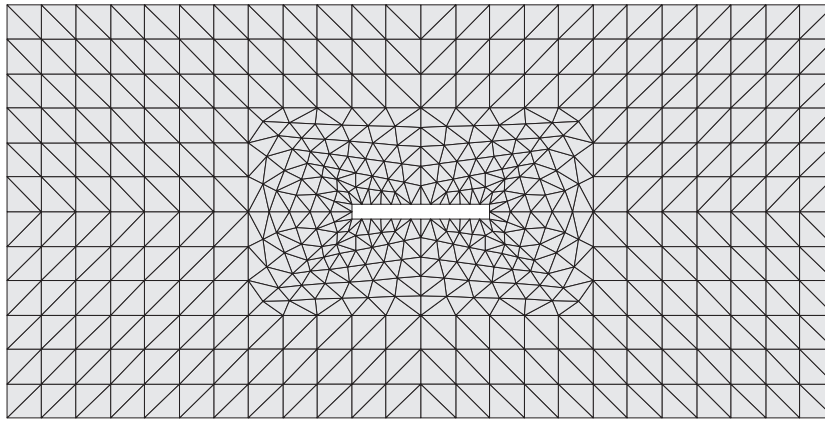


Figure 8-8: Example of a BQ card with an inhomogeneous segmentation

8.2.5 BT Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
BT		S_1	S_2	S_3			R_1	R_2	R_3					
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

Using this card three points are joined to form a triangle. This triangle will then be subdivided further into triangular elements. In figure 8-9 a sketch is shown.

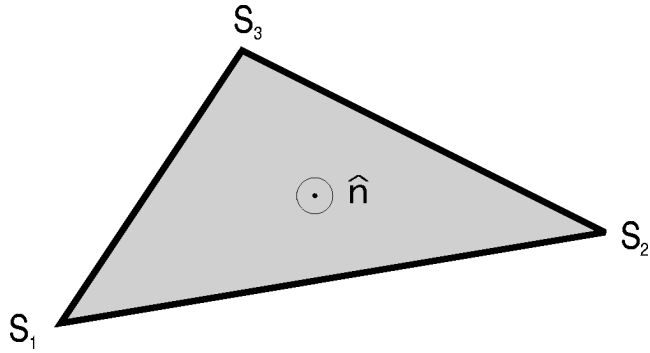


Figure 8-9: Sketch illustrating the use of the BT card

Parameters:

- S_1 Name of the first point of the triangle.
- S_2 Name of the second point of the triangle.
- S_3 Name of the third point of the triangle.
- R_1 Normally the triangle is segmented according to the triangle edge length specified with the IP card. (This is the default if the parameter R_1 is not specified.) However, it is often desirable to have an inhomogeneous segmentation, for example in the transition from a finely segmented region to a region with coarser segmentation. If set, the parameter R_1 specifies the triangle edge length along the edge S_2–S_3 (i.e. the side opposite S_1). R_1 is in m, and is affected by the SF card scaling factor.
- R_2 Similarly to R_1, but applies to the edge S_3–S_1 (opposite S_2).
- R_3 Similarly to R_1, but applies to the edge S_1–S_2 (opposite S_3).

The three points need to be defined previously with DP cards.

The direction of the normal vector ($\hat{n}$ of the subdivided triangles is determined by the right hand rule through all the corners. This direction only has meaning when used with the Physical Optics (PO card) or with dielectrics (ME card).

Examples of BT card usage:

Through the use of the following commands the triangle in figure 8-10 will be created.

```
**
IP                                        0.2
DP   A                          0.0       0.0       0.0
DP   B                          1.0       0.0       0.0
DP   C                          0.8       0.0       1.3
BT   A    B    C
EG
EN
```

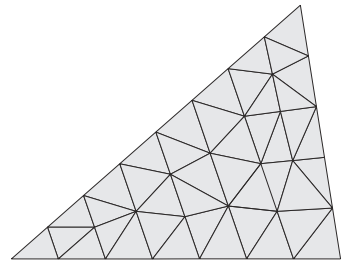


Figure 8-10: Example of a BT card

The triangle with inhomogeneous segmentation in figure 8-11 is created with

```
** non-uniform segmentation of a triangle
DP   A                          0         0         0
DP   B                          5         0         0
DP   C                          2         3         0
#l = 0.3
IP                                        #l
BT   A    B    C                3*#l      3*#l      1*#l
EG
EN
```

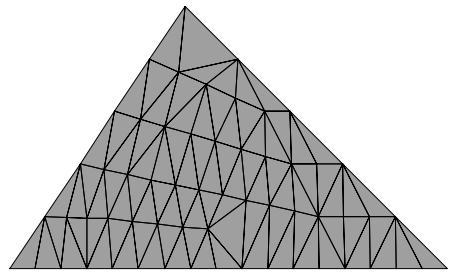
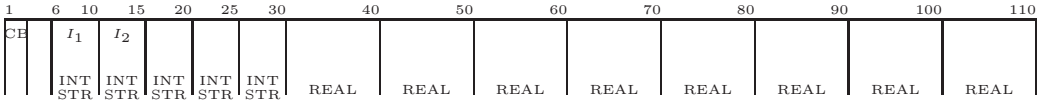


Figure 8-11: Example of a BT card with inhomogeneous segmentation

8.2.6 CB Card



This card is used to change or reassign the labels assigned to points, segments, triangles, cuboids, polygons, tetrahedral elements, etc. This is especially useful when more labels are created by using symmetry (SY card) or transformation (TG card) and, for example, edges or wedges in the PO area are considered.

Parameters:

- I_1 Old label.
- I_2 New label.

All structures with the label I_1, created and/or imported before processing the CB card, are assigned the new label I_2. Structures created after the CB card are not affected.

8.2.7 CL Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
CL	S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4	R_5				
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

This card can create an arc consisting of segments. Alternatively, this card may be used to create an arc, which forms the border to the PO region. See figure 8-12.

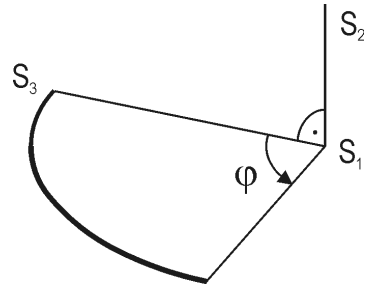


Figure 8-12: Sketch illustrating the use of the CL card

Parameters:

- S_1 Name of the centre of the circle.
- S_2 Name of the point perpendicular to the plane in which the circle lies and above its centre.
- S_3 Name of the starting point of the arc.
- S_4 If this field is empty, then an arc consisting of segments is generated. If there is an entry in this field then an arc, that forms the border to the PO region, is generated. S_4 is then the label of the bordering triangles.
- R_1 The subtended angle φ in degrees. The direction is in the positive sense around the S_1–S_2 axis.
- R_2 Maximum length of the segments that make up the arc (in m) (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.
- R_3 Normally the wire radius is set with the IP card. Setting R_3 overrides this radius for the current arc without affecting the default for later segments. (R_3 is in m and is affected by the SF card scaling factor.)
- R_4 By setting the parameter R_4 it is possible to create a wire with a tapered radius. The parameter R_3 must also be set, R_3 then specifies the radius at point S_3 while R_4 specifies the radius at the other end.

R_5 If this parameter is empty or is set to 1, a circular wire are is created. The parameter R_5 may be used to generate an elliptical arc (within reasonable limits). $R_5 = \frac{b}{a}$ gives the ratio of the two half axes of the ellipse, where a is the distance S_1–S_3.

Quite often modelling the geometry of an arc requires shorter segments than those used for straight wires. Thus the maximum segment length specified with the IP card can be overridden along the arc by setting R_2.

The radius of the arc is given by the distance between the points S_1 and S_3.

Examples of CL card usage:

The following commands creates the wire arc shown in figure 8-13.

```
**
IP                              0.01                    0.5
DP   A                          0.0         0.0         0.0
DP   B                          0.0         0.0         1.0
DP   C                          0.3         0.0         0.0
CL   A    B    C                270.0       0.1
EG
EN
```

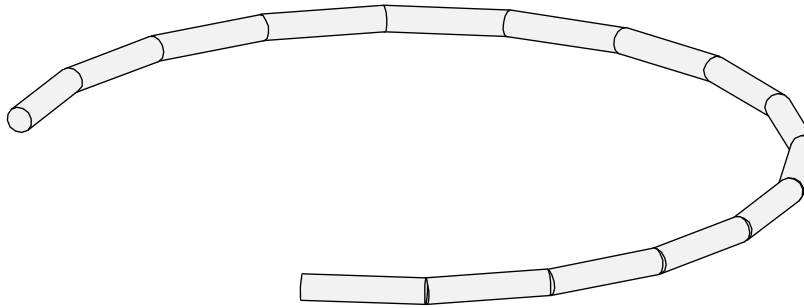


Figure 8-13: Example of a CL card

The following commands creates the tapered wire arc shown in figure 8-14.

```
**
IP                              0.01                    0.5
DP    A                         0           0           0
DP    B                         0           0           1
DP    C                         0.3         0           0
CL    A     B     C             270         0.1         0.005       0.03
EG
EN
```

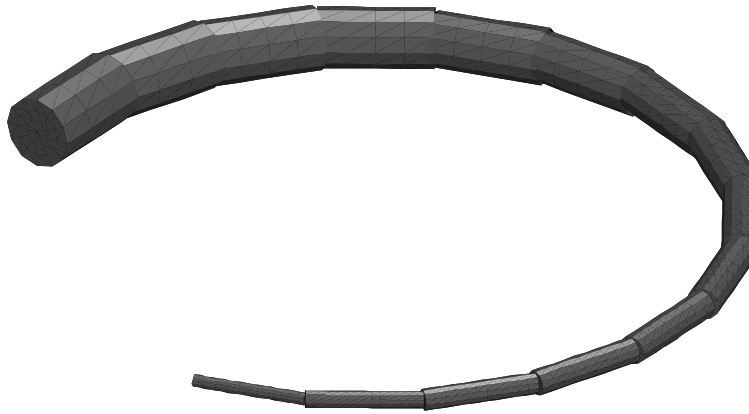
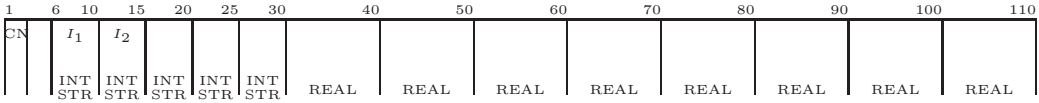


Figure 8-14: Example of a CL card with tapered wire radius

8.2.8 CN Card



This card is used to reverse the normal direction of previously created triangles and/or polygons, for example after importing CAD data. The normal direction is important in some cases, such as when defining dielectric surfaces.

Parameters:

I_1 0: Reverse the normal of all triangles with label I_2.
1: Reverse the normal of all polygons with label I_2.
2: Reverse the normal of the triangle with absolute number I_2.
3: Reverse the normal of the polygon with absolute number I_2.

I_2 The label or absolute element number, depending on the value of I_1.

For triangles, the normal vector is reversed by interchanging corners 1 and 3. For polygons the first point remains as is, but the corner points are listed in the opposite direction.

The CN card changes the normal of the affected triangles, but it does not change the settings of the ME card (which medium is on which side of the triangle as determined by the normal vector). For example, triangles defined after the card `ME 5 2` must have their normal vectors pointing from medium 5 to medium 2. Thus reversing the normal effectively change which medium lies on which physical side of the triangle.

8.2.9 DK Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
DK	S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4	R_5	R_6	R_7		
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

With this card a dielectric or magnetic eighth of a sphere, consisting of smaller cuboids, can be created.

Parameters:

- S_1 Name of the centre of the sphere.
- S_2, S_3, S_4 The three directions S_1–S_2, S_1–S_3 and S_1–S_4 form the border of the eighth of the sphere. All of them have to have the same length (sphere's radius) and be perpendicular to each other.
- R_1 Maximum side length of cuboids along the curved edge in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.

Only applicable to a dielectric sphere:

- R_2 Relative dielectric constant ε_r of the dielectric sphere.
- R_3 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the sphere.
- R_4 The density ϱ in $\frac{\mathrm{kg}}{\mathrm{m}^3}$ of the sphere. This parameter is only required to calculate the SAR (specific absorption rate), but it is compulsory, i.e. it must be defined.
- R_7 The electric loss tangent, $\tan(\delta)$

Only applicable to a magnetic sphere:

- R_5 Relative permeability μ_r of the magnetic sphere.
- R_6 The magnetic loss factor, $\tan(\delta_\mu)$

No dielectric bodies with surface equivalence principle and volume equivalence principle (using cuboids) can be used at the same time.

Example of DK card usage:

Using the following commands an eighth of a sphere as shown in figure 8-15 is created.

```
**
IP                                                              0.1
DP    A                           0.0         0.0         0.0
DP    B                           0.4         0.0         0.0
DP    C                           0.0         0.4         0.0
DP    D                           0.0         0.0         0.4
DK    A     B     C     D         0.06        4.0         0.0         1000.0
EG
EN
```

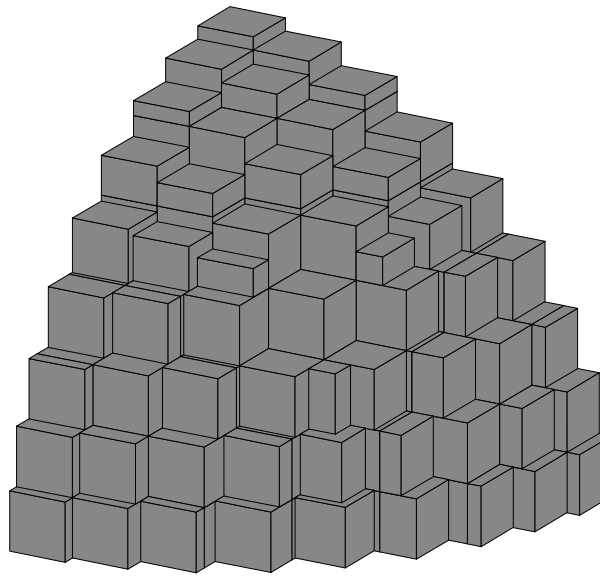
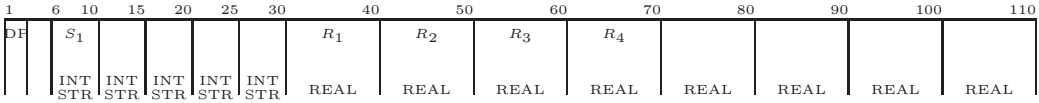


Figure 8-15: Example for the DK card

8.2.10 DP Card



With this card points in space are defined. To avoid ambiguity each point is assigned a name (a 5 character string). In the other commands (e.g. BL card) the points are referred to by their names.

Parameters:

S_1	Name (maximum of 5 characters) of the point.
R_1	x coordinate of the point in m (is scaled by the SF card).
R_2	y coordinate of the point in m (is scaled by the SF card).
R_3	z coordinate of the point in m (is scaled by the SF card).
R_4	The weight of the control point when this point is used with the NU card (NURBS surfaces). When this field is empty the default is 1.

In addition to its coordinates, each point is also assigned the current label (see LA card), so that a group of points can be selected by label, for example when moving points with the TP card.

In an exception to the rule that all geometry cards must appear before the EG card, this card (as well as the TP card) can be used after the EG to define points for use in the AP card.

When defining or using node names, simple variable names of the form `A#i` are allowed. The algorithm is that if a hash sign is found in a node point name, this hash sign and everything that follows is interpreted as a variable string, evaluated and rounded to the nearest integer. Thus if we have `#k=15` and use or define a point `P#k` then this is equivalent to using `P15` as point name. The length of the node name string (before and after expansion) is still limited to 5 characters.

For instance, it would now be possible to define the points P1 to P20 inside a loop.

```
!!for #k = 1 to 20
DP   P#k           .....
!!next
```

8.2.11 DZ Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
DZ	S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8	
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

With this card a dielectric cylinder, consisting of dielectric cuboids, can be created.

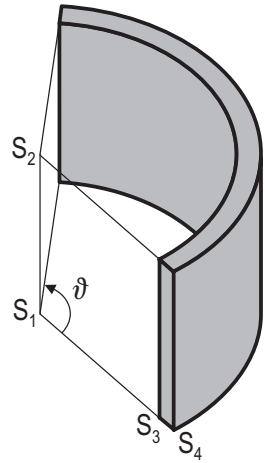


Figure 8-16: Sketch illustrating the use of the DZ card

Parameters:

- S_1 Name of the begin point of the cylinder's axis.
- S_2 Name of the outside of the cylinder.
- S_3 Name of a point that lies on the inside of the shell (see figure 8-16).
- S_4 Name of a point that lies on the outside of the shell (see figure 8-16).
- R_1 Angle φ in degrees of the cylindrical segment.
- R_2 Maximum edge length of the cuboids along the arc in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.

Only applicable to a dielectric cylinder:

- R_3 Relative dielectric constant ε_r of the cylinder.
- R_4 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the cylinder.
- R_5 The density ϱ in $\frac{\mathrm{kg}}{\mathrm{m}^3}$ of the cylinder. This parameter is only required to calculate the SAR (specific absorption rate), but it is compulsory and must always be defined.
- R_8 The electric loss tangent, $\tan(\delta)$

Only applicable to a magnetic cylinder:

R_6 Relative permeability μ_r of the magnetic cylinder.

R_7 The magnetic loss factor, $\tan(\delta_\mu)$

No dielectric bodies with surface equivalence principle and volume equivalence principle (using cuboids) can be used at the same time.

Example of DZ card usage:

Using the following command the cylindrical segment, as shown in figure 8-17, is created.

```
**
IP                                                          0.3
DP   A                          0          0          0
DP   B                          0          0          0.4
DP   C                          0.95       0          0
DP   D                          1          0          0
DZ   A    B    C    D          90         0.1        3          0          1000
EG
EN
```

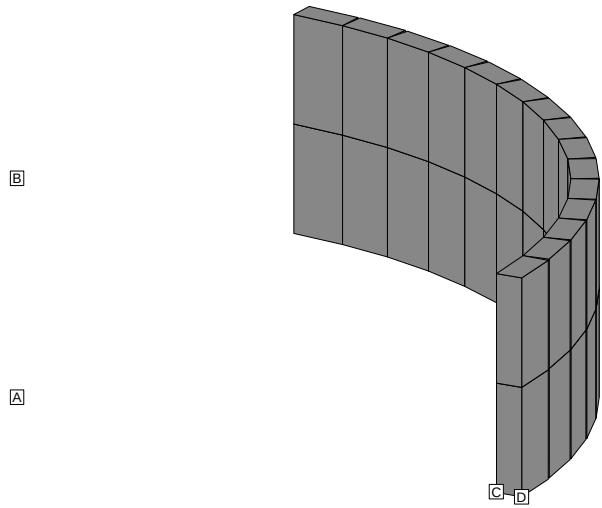


Figure 8-17: Example for the DZ card

8.2.12 EG Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
EG		PS1	PS5	NO CHECK	WMETHOD	USE_EDG	EPSENT	EPSR	MUER	SIGMA	TAND MUE	TAND EPS		
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card indicates the end of the geometrical input. It is essential that this card is used.

Parameters:

PS1
- 0: The geometric data of the segments and surface elements is written to the output file, but not to any additional files.
- 1: No geometric data is written, reducing the output file size.
- >1: When PS1 > 1, a bitwise and is used as follows
  - 1: write geometry information to the `*.out` file,
  - 2: write geometry information to a NASTRAN file,
  - 4: write geometry information to a STL file.

  For example, geometry output to `*.out`, NASTRAN and STL files is requested with PS1 = 7; NASTRAN only is requested with PS1 = 2.

PS5
- -1: No messages are sent to the standard output device (usually the screen).
- 0: Warnings, errors and messages that indicate the program's progress are sent to the standard output device.
- 1: Warnings, errors and explicit messages about the program's progress are sent to the standard output device. This option is useful during program development and debugging and thus is only available in the superuser mode.

NOCHECK
- 0: Normal procedure with verification of the geometry.
- 1: Verification of the geometry is switched off (see comment below).

WMETHOD
- 0: Normal version of FEKO.
- 1: Activates a special version of FEKO for extremely low frequencies, LFFEKO, (described in section 12).
- 2: Activates a special version of FEKO that uses the FMM (fast multipole method).
- 3: Activates a special version of FEKO that uses the FSSMM (faster single stage multipole method).
- 4: Activates a special version of FEKO that uses the MLFMM (multilevel fast multipole method).

USE_EDG
- 0: Normal run.
- 1: A `*.edg` file is used to store the connectivity information of the triangles. Binary format is used to keep the file size small. The `*.edg` file is read if it is present; and created — such that it may be read at a later run — if not. This is used to reduce the geometry set up computation time, especially for big models on large parallel computers.
- 2: Same as USE_EDG=1, but use an ASCII formatted version of the `*.edg` file. This can be copied between platforms, (e.g. when preparing models on a PC and running FEKO on a workstation) but the file can become quite large.

EPSENT — Limit separation for which two points in space are considered identical. For an exact description see below. If this parameter is not specified, it is set internally to 10^{-6} m. In most cases this value is sensible. If scaling is done with the SF card, this value is also scaled. It is recommended to specify EPSENT only if FEKO demands it, i.e. if a warning or an error message is given. Otherwise the default should be used.

EPSR — The relative dielectric constant ε_r of the homogeneous medium in which all structures are contained. When this field is empty ε_r is set to 1.

MUER — The relative permeability μ_r of the homogeneous surrounding medium. When this field is empty μ_r is set to 1.

SIGMA — The conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the homogeneous surrounding medium. When this field is empty σ is set to 0.

TANDMUE — Magnetic loss factor $\tan\delta_\mu$ of the homogeneous surrounding medium (the complex permeability is then given by $\mu = \mu_0\mu_r\,(1 - j\tan\delta_\mu)$).

TANDEPS — Electrical loss factor $\tan\delta$ of the homogeneous surrounding medium. This is an alternative way to specify the conductivity to which it is related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$.

The exact meaning of the parameter `EPSENT` is the following: The program PREFEKO creates a `*.fek` file, in which all the triangles and segments are described by their corner points. Due to rounding errors it is possible that, for example, end points of connecting segments do not coincide. When searching for nodes, an ohmic connection is made when the difference is smaller than the parameter `EPSENT`.

FEKO automatically checks for typical user errors that have been observed in the past. Examples of errors are connecting a wire segment to the middle of another wire, where the connection points do not coincide (see figure 2-2), or connecting surfaces that have different segmentation along the common edge (see figure 2-4). Such errors are detected

if the parameter NOCHECK is not set to 1. The error detection routine should always be used. However, if the same geometry is to be used a number of times, the error detection can be disabled by setting NOCHECK=1.

If the surrounding medium is not vacuum, one can set the material parameters with the EG card as shown above. Alternatively the parameters of the surrounding medium can be set with the GF card which offers greater flexibility. For example, the GF card can be used to set the material parameters (as an arbitrary function of frequency) inside a frequency loop which is not possible with the EG card.

8.2.13 EL Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
EL		S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4	R_5			
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card a section of an ellipsoid can be generated (see figure 8-18).

Parameters:

- S_1 Name of the point at the centre of the ellipsoid.
- S_2 Name of a point, in the direction $\vartheta = 0$ in elliptical coordinates. The distance of the two points S_1 and S_2 determines half the axis of the ellipsoid in this direction.
- S_3 Name of a point in the direction $\varphi = 0$ in elliptical coordinates. The distance of the two points S_1 and S_3 determines half of the axis of the ellipsoid in this direction.
- S_4 Name of a point in the direction of the third coordinate, i.e. the distance $S_4 - S_1$, $S_3 - S_1$ and $S_2 - S_1$ must be perpendicular. The distance of the two points S_1 and S_4 determines half of the axis of the ellipsoid in this direction.
- R_1 Begin angle ϑ_a in degrees of the ellipsoid.
- R_2 Begin angle φ_a in degrees of the ellipsoid.
- R_3 End angle ϑ_e in degrees of the ellipsoid.
- R_4 End angle φ_e in degrees of the ellipsoid.
- R_5 Maximum edge length of the triangles along the curved edge in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.

Note that the angles ϑ and φ are defined in an elliptical, rather than in a spherical coordinate system. For a Cartesian coordinate system with origin S_1, the x axis in direction of S_3, the y axis in the direction of S_4 and the z axis in the direction of S_2, a point on the surface of the ellipsoid is given as

$$\vec{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a \sin\vartheta \cos\varphi \\ b \sin\vartheta \sin\varphi \\ c \cos\vartheta \end{bmatrix}$$

where the lengths a, b and c are the lengths of the ellipsoid's three half-axes. (For example the length a is the distance between the points S_3 and S_1).

The normal vector of the generated triangles always points outwards. The algorithm used for the segmentation can fail if the ratio of the half-axis is too extreme, for example if the longest half-axis is a factor 100 longer than the shortest. It is strongly advised to check the geometry with WinFEKO.

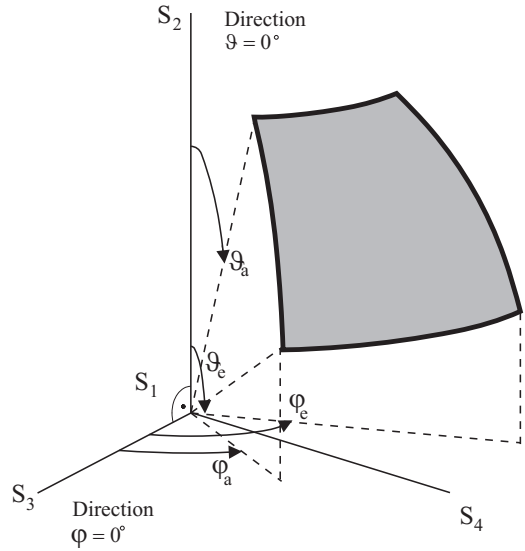


Figure 8-18: Sketch illustrating the use of the EL card

Example of EL card usage:

The following commands generate the eighth of an ellipsoid, as shown in figure 8-19.

```
**
IP                                          0.25
DP   A                          0           0          0
DP   B                          0           0          1.3
DP   C                          1.0         0          0
DP   D                          0           0.7        0
EL   A    B    C    D          0           0          90         90         0.25
EG
EN
```

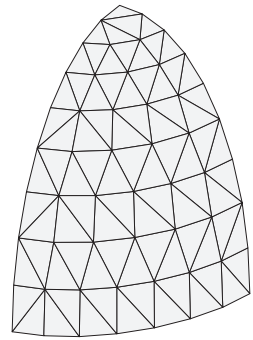
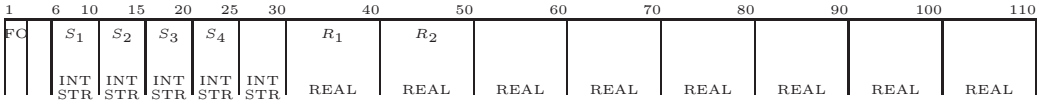


Figure 8-19: Example for the EL card

8.2.14 FO Card



With this card an area is defined in which the surface current density is an approximation according to the Fock theory.

Parameters:

- S_1 Type of Fock area:
  1: Ideal conducting cylinder.
  2: Ideal conducting sphere.
- S_2 Label of the metallic triangles that form the surface of the Fock area (e.g. the surface of the cylinder).
- S_3 Centre of the sphere (for $S_1 = 2$) or begin point of the axis of the cylinder (for $S_1 = 1$).
- S_4 Only defined for $S_1 = 1$, the name of the end point on the axis of the cylinder.
- R_1 Type of process for the Fock currents:
  0: Method by Daniel Bouche.
  1: Method by Louis N. Medgyesi–Mitschang.
- R_2 0: Usual option where the coupling between the MoM and the Fock regions is considered.
  1: The coupling between the MoM and Fock regions is neglected, so that there is no feedback by which the Fock currents may influence the current distribution in the MoM region. This option, which is particularly applicable when the MoM and Fock regions are not in close proximity, should result in a considerable reduction in computational effort and storage space.

The radius of the cylinder or sphere, does not have to be defined. It is determined by the distance of the metallic triangles with the label S_2 to the axis or centre.

The cylinder Fock currents can also be applied to cones (KK card, approximated by a staircase construction of cylinders) and sections of a torus that resembles a cylinder (TO card).

Although the FO card is strictly only applicable to spherical and cylindrical surfaces, it is often a good approximation on conical and toroidal surfaces.

8.2.15 HE Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
HE	S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4	R_5				
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

With this card a helical coil, consisting of wire segments can be created. A sketch is shown in figure 8-20.

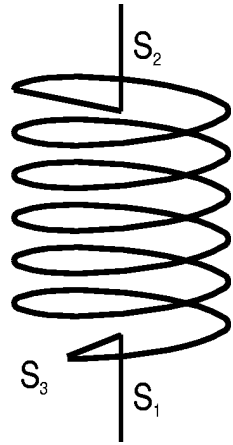


Figure 8-20: Sketch illustrating the use of the HE card

Parameters:

- S_1 Name of the begin point of the coil's axis.
- S_2 Name of the end point of the coil's axis.
- S_3 Name of the begin point of the windings.
- S_4 0: Create connections from the two ends of the coil to the axis (at points S_1 and S_2). See also left side of figure 8-21.
  1: The connections are not generated, this means that the point S_3 is a connection point. See also the right side of figure 8-21.
- R_1 Amount of windings. (If negative a left handed coil is created.)
- R_2 Maximum length of the segments, that are used for the windings in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.
- R_3 Normally the wire radius is set with the IP card. Setting R_3 overrides this radius for the current helix without affecting the default for later segments. (R_3 is in m and is affected by the SF card scaling factor.)

R_4 A helix with a tapered radius can be created by setting *both* R_3 (which then specifies the radius at the start point S_1) and R_4 (which specifies the radius at the end point S_2). The segments connecting to the axis (see S_4) are not tapered and have radii R_3 and R_4 respectively.

R_5 If this parameter is empty or is set to 1, a helix with a circular cross section is created. Setting R_5 allows (within reasonable limits) generating a helix with an elliptical cross section. $R_5 = \frac{b}{a}$ gives the ratio of the ellipse's two half axes, where a is the distance S_1–S_3.

Quite often modelling the geometry of the coil requires shorter segments than those used for straight wires. Thus the maximum segment length specified by the IP card can be overridden along the arc by setting R_2.

The windings are generated between the two points S_1 and S_2, that lie on the axis. The radius of the coil is defined by the distance between the points S_1 and S_3. For elliptical cross sections this is the length of one half axis and the other one is R_5 times this length.

Example of HE card usage:

The two coils shown in figure 8-21 are created with the following commands.

```
**
IP                                  0.02                0.5
DP    A1                            0.0       0.0       0.0
DP    B1                            0.0       0.0       1.0
DP    C1                            0.3       0.0       0.0
HE    A1    B1    C1    0           4.5       0.2
DP    A2                            0.0       1.0       0.0
DP    B2                            0.0       1.0       1.0
DP    C2                            0.3       1.0       0.0
HE    A2    B2    C2    1          -4.5       0.2
EG
EN
```

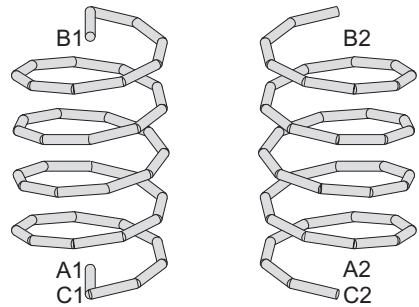


Figure 8-21: HE card examples — the coil on the right has a negative R_1.

8.2.16 IN Card

This card is used to include external files. These files may be other `*.pre` files (which are included as if they were part of the master file) *or* geometrical data files containing wire segments, triangles, quadrangles, tetrahedral volume elements and/or polygonal plates (in FEMAP neutral, an ASCII format, NASTRAN, meshed AutoCAD `*.dxf`, NEC model, Concept model or STL files).

The syntax for the IN card is somewhat different from the other cards. The fields are separated by an arbitrary number of spaces and the general format is

```
IN  FLAG  x  "filename.ext"  LABEL  LABEL_END  scaling=...;
```

where the integer parameter FLAG specifies the type of file. The filename is specified within quotation marks and may contain directory names as well, for example, `IN  "..\myfiles\include.inc"`. Both \ and / are allowed on Windows and UNIX systems. These parameters are required for all file types.

Label selection: LABEL and LABEL_END are optional integer parameters. If they are not present, the entire model is included. If only LABEL is present only structures with this label will be included. If both fields are present all structures with labels in the range LABEL to LABEL_END (inclusive) will be imported. (Note that layers/ properties/tags in the model files are all converted to FEKO labels. See the description for the different file types below.) Typical examples, here for FEMAP neutral files, are

```
IN  1  "part_1.neu"
IN  1  "part_2.neu"  5
IN  1  "part_3.neu"  7  9
```

The first line will include all structures in the FEMAP model, the second only those on layer 5 and the last line all structures on layers 7, 8 and 9. (FEMAP layers are converted to FEKO labels.) Label selection is not supported for `*.pre`, Concept and STL files.

Type selection: The type selection flag `x` is optional for certain input formats (FEMAP, ASCII, NASTRAN, AutoCAD, PATRAN) and is required for Concept models. (For `*.pre` and STL files it is not supported.) This parameter allows selecting only certain types of structures where the parameter `x` has the following meaning:

1	Wire segments
2	Surface triangles
4	Polygonal plates
8	Tetrahedral elements
16	Points
32	Quadrangles (subdivided into two triangles during import)
64	Points, but only those used by the geometry

It is possible to add these options, i.e. for `x = 1+2 = 3` both wire segments and triangles shall be imported (provided they also satisfy the label criterion), but no polygonal plates, tetrahedral volume elements or points. (Quadrangles are sometimes grouped with triangles and sometimes considered separately.) The default, when the parameter `x` is not specified, is to include all structures (but not points, these must be requested explicitly) that satisfy the label criterion. Note that some of these options may not be supported for a given file type. (See the description for the different file types below.)

Scaling: An optional constant scaling factor can be applied to the imported geometry. This is necessary, for example, if separate CAD files with different units must be imported, or if the `*.pre` is, for example, created using mm while the CAD file is constructed using inches as unit. Scaling is specified by adding the character string `scaling=...;` (it ends with a semicolon) at the end of the card. The syntax is then, for example,

```
IN  2  4   "filename.dat"        scaling=0.001;
IN  2      "filename.dat"  7     scaling=0.001;
IN  1      "filename.neu"  0  3  scaling=0.001;
```

Variables and functions may be used in the expression for the scaling factor, for example,

```
IN  1     "filename.neu"     scaling=3/5;
IN  4     "filename.dxf"  7  scaling= #my_scal;
IN  3  1  "filename.nas"     scaling= (1.0-0.2) / (#x_var + sin(0.71));
```

It should be noted that the scaling factor specified with the IN card is applied in addition to any scaling factor that may be set with the SF or TG cards. When reading a `*.pre` file (FLAG=0) it is not possible to add a scaling factor to the IN card. In this case the TG card must be used if the global (SF card) scaling option is not sufficient.

The parameter FLAG determines the type of file

- **FLAG=0 (Read a `*.pre` file)**

  This allows large `*.pre` files to be split into several parts. It is, for example, particularly useful when the same geometry is used in different files. The syntax is

  ```
  IN   0   "head.pre"
  IN   0   "body.pre"
  ```

  and no other parameters are supported. The cards in the included files are processed as if they were part of the main file. Therefore points and labels defined in the included file remain valid in the remainder of the main file.

  Note that the file must have the syntax of a `*.pre` file, but the extension can be arbitrary. It is suggested to use, for example, `*.inc` (for include) to clearly distinguish between main `*.pre` files (that are input for FEKO) and include files.

- **FLAG=1 (Read a `*.neu` FEMAP neutral file)**

  This card is used to import models generated by the commercial CAD / meshing program FEMAP. (The models must be exported from FEMAP in the `*.neu` FEMAP neutral file format.) The syntax is then

```
IN  1  "part_1.neu"
IN  1  "part_2.neu"  5
IN  1  7  "part_3.neu"  7  10  scaling=0.001;
```

  The label selection uses the FEMAP layer numbers which are converted to FEKO labels. Note that scaling is supported. The type selection parameter `x` is supported and may have the following values

1	Wire segments
2	Surface triangles
4	Polygonal plates
8	Tetrahedral volume elements
16	Node points
32	Quadrangles (divided into triangles)

  Wires must be meshed into elements which are imported as segments, surfaces into triangles or quadrangles which are imported as FEKO triangles, and boundary surfaces are imported as polygons. The boundary surface must be bordered with *line* curves rather than *edge* curves.

  The user can also elect to import points from the `*.neu` file. All points defined as such in FEMAP are then available in PREFEKO as points (as if they were defined by DP cards) of the form `Pxxx` where `xxx` is the point ID in FEMAP. This may be used, for example, when attaching additional structures to a geometry partly created in FEMAP. In addition, the coordinate values of the point are available as variables in PREFEKO. For example, the variables `#p1234x`, `#p1234y` and `#p1234z` give the coordinates of the FEMAP point with ID 1234. Note that points are not included by default.

  It should be remembered that it is not possible to specify a wire radius in FEMAP. Thus the wire radius must be specified by an IP card preceding the IN card. Similarly, when specifying the surface of a dielectric, the IN card must be preceded with the correct ME card (completely analogous to the case without FEMAP).

  The following illustrates the use of the IN card to include FEMAP neutral files.

```
** Set wire radius
IP    ...

** Include a metallic structure (all structures in the model)
IN   1   "phone.neu"
```

```
** Translate or rotate the above geometries
TG    ...

** Include an additional dielectric structure (only layers 3 to 5)
ME   1   0
IN   1   "head.neu"  3  5

** Some additional geometric data
DP   A  ...
DP   B  ...
BL   A  B
...
```

WinFEKO should be used to verify the included geometry.

- **FLAG=2 (Read geometry from an ASCII data file)**

  With the syntax

  ```
  IN  2  "part_1.dat"
  IN  2  "part_2.dat"  5
  IN  2  7  "part_3.dat"  7  10  scaling=0.1;
  ```

  the IN card reads the geometry data from a data file, whose structure is described below. Both label and type selection and scaling are supported. The type selection parameter `x` may have the following values

  1 Wire segments
  2 Surface triangles
  4 Polygonal plates
  8 Tetrahedral volume elements

  Dielectric triangles or metallic triangles which form the surface of a dielectric, are created by preceding the IN card with the appropriate ME card. (In exactly the same way as is the case without the IN card.)

  The data of the segments, triangles and polygonal plates are given in an ASCII file, formatted as shown below. There is no need to adhere to specific columns, the data fields merely have to be separated by one or more spaces.

  ```
  nk   nd   ns   np   nt
  x(1)   y(1)   z(1)  (String_name)
  x(2)   y(2)   z(2)  (String_name)
  ...

  x(nk)  y(nk)  z(nk)  (String_name)
  ```

```
d1(1)  d2(1)  d3(1)  0  (Label)
d1(2)  d2(2)  d3(2)  0  (Label)
...

d1(nd) d2(nd) d3(nd) 0  (Label)
s1(1)  s2(1)  0      0  (Label)
s1(2)  s2(2)  0      0  (Label)
...

s1(ns) s2(ns) 0      0  (Label)
nnp(1)   p1(1)  p2(1)  p3(1)  ...   (Label)
nnp(2)   p1(2)  p2(2)  p3(2)  ...   (Label)
...

nnp(np)  p1(np) p2(np) p3(np) ...   (Label)
t1(1)  t2(1)  t3(1)  t4(1)  (Label)
t1(2)  t2(2)  t3(2)  t4(2)  (Label)
...

t1(nt)  t2(nt)  t3(nt)  t4(nt)  (Label)
```

The meaning of the above is:

nk	Number of nodes
nd	Number of triangles
ns	Number of segments
np	Number of polygonal plates
nt	Number of tetrahedral volume elements (defaults to 0 if not specified)
$x(i)$	x coordinates of node i in m (is scaled by the SF card)
$y(i)$	y coordinates of node i in m (is scaled by the SF card)
$z(i)$	z coordinates of node i in m (is scaled by the SF card)
$d1(j)$	Number (index) of the first vertex of triangle j
$d2(j)$	Number (index) of the second vertex of triangle j
$d3(j)$	Number (index) of the third vertex of triangle j
$s1(k)$	Number (index) of the starting point of segment k
$s2(k)$	Number (index) of the end point of segment k
$nnp(m)$	Number of corner points in polygon m
$p1(m)$	Number (index) of the first corner of polygon m
$p2(m)$	Number (index) of the second corner of polygon m
$p3(m)$	Number (index) of the third corner of polygon m
...	Number (index) of the additional corners of polygon m

$t1(n)$	Number (index) of the first node point of tetrahedra n
$t2(n)$	Number (index) of the second node point of tetrahedra n
$t3(n)$	Number (index) of the third node point of tetrahedra n
$t4(n)$	Number (index) of the fourth node point of tetrahedra n
String_name	Optional string name of the point. It must be a string of up to 5-characters, similar to the point name of the DP card. If a point is named, it can be used in any card following the IN card.
Label	Specifying the label as the last parameter of any structure is optional. If no label is specified, the value defined at the last LA card will be used. Note that if a label or range of labels is specified (with parameters after the filename), this LA card label will be used to determine if a structure is included or not.

The radius of segments must be specified by an IP card before the IN card. It is recommended to check the geometry with WinFEKO.

Example:

The structure in figure 8-22, consisting of 5 node points and 3 triangles with label 7 (no segments or polygonal plates), may be generated with the following data file

```
5  3  0  0
3.0  0.0  1.0
4.0  2.0  1.0
2.5  3.0  2.5
0.0  3.0  4.0
1.0  0.0  3.0
1  2  3  0  7
1  3  5  0  7
3  4  5  0  7
```

- **FLAG=3 (Read geometry from a NASTRAN file)**

PREFEKO supports importing column based NASTRAN files. Only the keywords GRID, CTRIA3, CQUAD4, CBAR, CROD and CTETRA for nodes, triangles, quadrangles (divided into two triangles along the shortest diagonal), segments and tetrahedral volume elements are processed. All other keywords are ignored. In this case the syntax is

```
IN  3  "part_1.nas"
IN  3  "part_2.nas"  5
IN  3  35  "part_3.nas"  7  10  scaling=25.4;
```

The label selection uses the NASTRAN properties which are converted to FEKO labels. Note that scaling is supported.

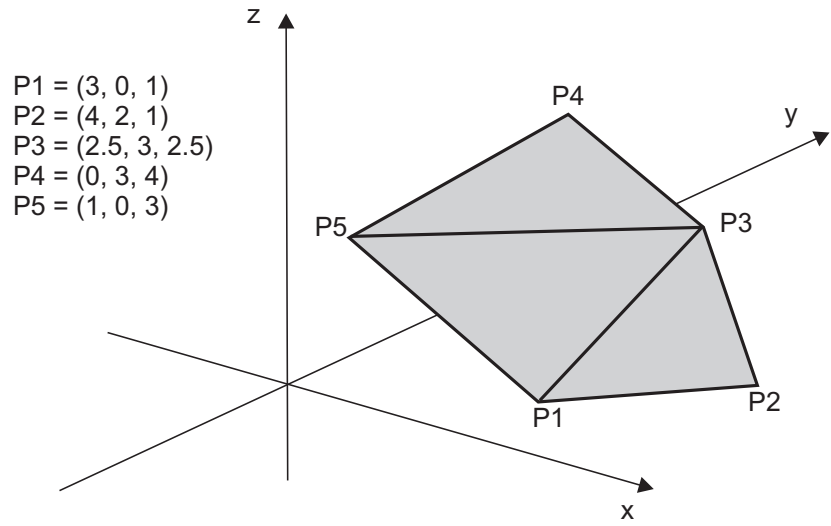


Figure 8-22: Example for IN card

The type selection parameter `x` is supported and may have the following values

1	Wire segments
2	Surface triangles
16	Points
32	Quadrangles (divided into triangles)
64	Points, but only those used by the imported geometry

As when importing `*.neu` files, the wire radius must be set with the IP card preceding the IN card, and an ME card must be used when specifying dielectric surfaces in the same way as when the IN card is not present.

The user can also import points from the NASTRAN file similar to importing points from FEMAP. The points defined in the NASTRAN file will then available in PREFEKO as points (as if they were defined by DP cards) of the form `Nxxx` where `xxx` is the index of the grid point. This may be used, for example, to attach additional structures to the geometry. In addition, the coordinate values of the point are available as variables in PREFEKO. For example, the variables `#n1234x`, `#n1234y` and `#n1234z` give the coordinates of the NASTRAN grid point with index 1234. Note that points are not included by default. Since grid points do not have an associated property, points are imported irrespective of their label.

Each line in the column based format consists of one keyword such as “GRID” starting in column 1. From column 9 onwards follow 9 input fields with a width of 8 characters each. Thus input field 1 uses columns 9 to 16, input field 2 columns 17 to 24 etc. The ninth (and last) input field 9 ends at column 80. Below is

a very simple NASTRAN example file consisting of a plate (property 1; subdivided into eight triangles) and a rod (property 2; subdivided into two segments).

```
|1       |9       |17      |25      |33      |41      |49      |49
ID XXXXXXXX,YYYYYYYY
CEND
BEGIN BULK
GRID            1             0.0     0.0     0.0
GRID            2         0.50000     0.0     0.0
GRID            3         1.00000     0.0     0.0
GRID            4             0.0 0.50000     0.0
GRID            5         0.50000 0.50000     0.0
GRID            6         1.00000 0.50000     0.0
GRID            7             0.0 1.00000     0.0
GRID            8         0.50000 1.00000     0.0
GRID            9         1.00000 1.00000     0.0
GRID           10         0.50000 0.50000 2.00000
GRID           11         0.50000 0.50000 1.00000
CROD            9       2       5      11
CROD           10       2      11      10
CTRIA3          1       1       4       5       8
CTRIA3          2       1       4       8       7
CTRIA3          3       1       5       6       9
CTRIA3          4       1       5       9       8
CTRIA3          5       1       1       2       5
CTRIA3          6       1       1       5       4
CTRIA3          7       1       2       3       6
CTRIA3          8       1       2       6       5
ENDDATA
```

It may be imported using the commands

```
IP                              0.001
IN   3   "geometry.nas"
```

For the node points FEKO also supports 16 character wide input fields. The keyword GRID in columns 1 to 4 is followed by a star and three spaces. The node ID is then in columns 9 to 24, the x coordinate in columns 43 to 56, y in columns 57 to 72 and z in columns 9 to 24 of the next line, for example

```
|1       |9                  |25    ...  |43              |57              |73      |81
GRID*                   1                 50.000000000   -18.480176926        1
*       1    23.222875595
GRID*                   2                 50.000000000   -18.480176926        2
*       2   -13.410394669
```

- **FLAG=4 (Read geometry from an AutoCAD `*.dxf` file)**

  This card allows importing `*.dxf` models. The `*.dxf` file must comply with the *release 12* DXF format specifications. It should contain *meshed* surfaces in the form of polylines (see the discussion below) and lines (that will be segmented by PREFEKO as discussed below).

  The syntax is similar to that of the ASCII, FEMAP and NASTRAN cases.

```
IN  4  "part_1.dxf"
IN  4  2  "part_2.dxf"  5
IN  4  3  "part_3.dxf"  7  10  scaling=0.1;
```

  Layers named `n` or `LAYER_n` (where `n` is an integer number) in the `*.dxf` model are converted to label `n` in FEKO. For all structures for which no label is defined in this format, the label specified with the last LA card preceding the IN card is used. (If no such LA card is in effect, the default is label 0.) This label is used in the label selection. As for ASCII, FEMAP and NASTRAN files, scaling is supported. In this case the type selection parameter `x` may have the following values

  1 Wire segments
  2 Surface triangles and quadrangles

  Presently there is no provision to import polygonal plates or cuboidal volume elements from `*.dxf` models. As for the other CAD models, dielectric triangles or metallic triangles which form the surface of a dielectric, are created by preceding the IN card with the appropriate ME card.

  PREFEKO only processes the geometry information in the section of the file between the keywords `ENTITIES` and `ENDSEC`. Segments are imported from blocks defined by the keyword `LINE`.

```
  0
LINE
  8
LAYER_01
   .
   .
 10
-0.0538
 20
0.0
 30
8.134
 11
5.110
 21
```

```
2.857
 31
0.0
  0
 ... (next keyword)
```

The group code 8 at some point below LINE indicates that the next line contains the layer name. In this case, the layer will be converted to label 1. The line will be imported and segmented if this label lies in the required range. (If not, PREFEKO will search for the next occurrence of LINE.) Next the x, y and z components of the start point follow the group codes 10, 20 and 30; and those of the end point follow the codes 11, 21 and 31.

Here the start and end points are (x, y, z) = (-0.0538, 0.0, 8.134) and (5.110, 2.857, 0.0) respectively. If any of the coordinate group codes is not present (such as in a 2D model), the related coordinate is set to zero. The block is terminated by the group code 0. The wire is segmented according to the maximum segment length specified by the IP card, and the segments also have the radius specified by this card.

Meshed surfaces are imported from blocks denoted with the keyword POLYLINE. This block contains the layer name (following the group code 8 as before; if there is no group code 8 before the first VERTEX, the label specified with the last LA card will be used) and a number of VERTEX structures. There can be an arbitrary number of VERTEX structures, but there should be at least four.

The POLYLINE structure is terminated by the keyword SEQEND.

```
  0
POLYLINE
  8
LAYER_02
  .
  .
VERTEX
  .
  .
VERTEX
  .
  .
VERTEX
  .
  .
  0
SEQEND
```

There are two types of vertices. The first type defines points is space

```
  0
VERTEX
  8
LAYER_02
   .
   .
 10
7.919192
 20
3.393939
 30
0.0
   .
   .
  0
 ... (next keyword)
```

where the x, y and z components of the point follow the group codes `10`, `20` and `30`. The layer information is ignored.

The second type of vertex is a "linker".

```
  0
VERTEX
  8
LAYER_02
   .
   .
 70
    128
 71
     4
 72
     2
 73
     1
 74
     3
  .
  .
  0
 ... (next keyword)
```

which defines a triangle or quadrangle by specifying the indices (starting from 1 in the order the non-linker vertices are specified) of the vertices which form its corner points. Vertices are defined as linkers by setting a value of `128` in the group code `70` field. For linker vertices the coordinates are ignored. Note that some old `*.dxf` versions do not contain linker vertices — they cannot be imported. (Usually they do not contain mesh information.)

The four integer numbers after the group codes `71`, `72`, `73` and `74` give the indices of corners of the triangle or quadrangle. (In the case of a triangle one of these is absent.) PREFEKO divides each quadrangle into two triangles along the shortest diagonal.

Note that the `*.dxf` specification does not require the group codes to be in any specific order. However, PREFEKO requires that the group codes `71`, `72`, `73` and `74` follow group code `70`.

If these extracts are taken from the file `geometry.dxf`, both the line (meshed as segments) and the two polyline triangles may be imported with the commands

```
#rad = 0.001
#seglen = 1
IP                              #rad                #seglen
IN   4    "geometry.dxf"   1  2
```

- **FLAG=5 (Read geometry from a NEC input file)**

  PREFEKO also supports importing wire geometry from NEC[6] models. Note that NEC models usually consist of wire grid surfaces and it would be more efficient to convert the models to FEKO surfaces, but this cannot be done automatically.

  In this case the syntax is

  ```
  IN  5  "filename.nec"
  IN  5  "filename.nec"  3
  IN  5  1  "filename.nec"  3   7  scaling=0.001;
  ```

  The label selection uses the NEC tags which are converted to FEKO labels. This applies to the tag when the element is defined. If the tag is modified after the inclusion (for example with the GM card) the elements with the modified tag are also included. Again scaling is supported. The type selection parameter `x` is also supported, but it may only have the value `1` for wire segments.

  The NEC import filter considers only the geometry cards CM, CE, GA, GW, GM, GR, GS, GX and GE. A warning is given if other cards are encountered. If the model contains multiple geometries only the first one is read.

[6] G.J. Burke and A.J. Poggio, "Numerical Electromagnetics Code (NEC) — Method of Moments," Lawrence Livermore Laboratory, January 1981.

- **FLAG=6 (Import of Concept geometry files)**

  With this option one may import Concept[7] geometry files. Since Concept uses two different files for wires and surface elements, the type selection parameter `x` is obligatory and determines the type of geometry file to be read:

```
IN  6   1  "concept_wire.dat"
IN  6  34  "concept_surface.dat"  scaling=1000;
```

  The selection parameter must be either 1 (for wire segments) or 34 (determined from 2+32 for surface elements — triangles and quadrangles). Other values and other binary additions are not allowed. Similar to the AutoCAD import, quadrangles are automatically subdivided (along the shorter diagonal) into two triangular patches. Again scaling is allowed.

  Since wires don't have a radius in the model files, the radius is specified with a preceding IP card. Likewise, the elements don't have labels, and the label as specified at the last LA card before the IN card is used. If there is no LA card, the label defaults to zero. Obviously, since no labels are present in the Concept files, a label selective import is not supported.

  As for the CAD models, dielectric triangles or metallic triangles which form the surface of a dielectric, are created by preceding the IN card with the appropriate ME card.

  The Concept files for wires is as follows

```
number_of_wires
x_start y_start z_start x_end y_end z_end   [number_of_wires times]
```

  where the first integer specifies the number of wires followed by the coordinates of the start and end point of each wire. The file is completely free format — the values are just separated by white space. The surface file is

```
number_of_nodes   number_of_patches
x y z                                       [number_of_nodes times]
p1 p2 p3 p4                                 [number_of_patches times]
```

  again using free format. The values `x`, `y` and `z` specify the node coordinates and `p1`, `p2`, `p3` and `p4` specify the corner nodes of the triangles (in this case `p4` is `0`) and quadrangles.

[7] A MoM code developed at the University of Hamburg–Harburg, Germany.

- **FLAG=7 (Import of triangular data from STL files)**

  PREFEKO can also import STL — both ASCII and binary — files. STL files support only triangular patches and these are all imported. Therefore the selection parameter `x` does not make sense and is not supported. Also, since the STL file makes no provision for any labels, label selection is not supported. The syntax is

```
IN  7  "file.stl"
IN  7  "file.stl"   scaling=0.001;
```

  Note that scaling is supported. An example of an ASCII STL file is

```
SOLID CATIA STL PRODUCT
  FACET NORMAL -4.602166E-01 -1.858978E-01 -8.681260E-01
    OUTER LOOP
      VERTEX    4.789964E-01 -8.440244E-01  2.878882E-01
      VERTEX    4.764872E-01 -8.439470E-01  2.892018E-01
      VERTEX    4.783065E-01 -8.414296E-01  2.876983E-01
    ENDLOOP
  ENDFACET
  FACET NORMAL -4.601843E-01 -1.859276E-01 -8.681367E-01
    OUTER LOOP
      VERTEX    4.764872E-01 -8.439470E-01  2.892018E-01
      VERTEX    4.761175E-01 -8.425569E-01  2.891001E-01
      VERTEX    4.783065E-01 -8.414296E-01  2.876983E-01
    ENDLOOP
  ENDFACET
ENDSOLID
```

  For the description of binary STL files, please see
  http://www.ennex.com/fabbers/StL.asp
  http://rpdrc.ic.polyu.edu.hk/old_files/stl_binary_format.htm

- **FLAG=8 (Read a CADFEKO model file)**

  This option is used to import models generated by CADFEKO. These models are already meshed. The syntax of the IN card is then

```
IN  8  "part_1.neu"
IN  8  "part_2.neu"  5
IN  8  7  "part_3.neu"  7  10  scaling=0.001;
```

  As for a number of import options, label selection is allowed. Also scaling is supported. The type selection parameter `x` is supported and may have the following

values

1 Wire segments
2 Surface triangles
8 Tetrahedral volume elements
16 Node points

If the parameter `x` is omitted, wire segments and surface triangles are imported, but node points are not.

Points can be imported similar to the case when importing `*.neu` files. All points defined in the CADFEKO model are then available in PREFEKO (as if they were defined with DP cards) with the names as defined in CADFEKO. This may be used, for example, to attaching additional structures to the geometry using PREFEKO cards. Unlike the case for FEMAP and NASTRAN the coordinate values are not defined as variables. If these are required, the user should use the `x_coord`, `y_coord` and `z_coord` functions.

WinFEKO should be used to verify the included geometry.

- **FLAG=9 (Read geometry from a PATRAN file)**

PREFEKO also supports importing PATRAN files. Only the following PATRAN neutral packet types are imported:

`01` Node data (all coordinates are interpreted in the global rectangular frame, local coordinate frames are not supported).
`02` Element data. The shapes 2 = bar, 3 = tri, 4 = quad and 5 = tet are allowed. Quadrangles are automatically subdivided into triangles along the shortest diagonal.
`99` End of file flag.

Other packet types are ignored. The syntax to import PATRAN models is

```
IN  9  "part_1.pat"
IN  9  "part_2.pat"  5
IN  9  35  "part_3.pat"  7  10  scaling=25.4;
```

The label selection uses the PATRAN material ID's which are converted to FEKO labels. Note that scaling is supported.

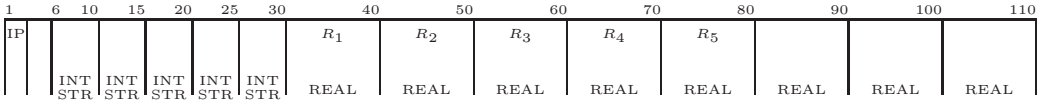
The type selection parameter `x` is similar to the case for NASTRAN files and may have the following values

1 Wire segments
2 Surface triangles
8 Tetrahedral volume elements
16 Points
32 Quadrangles (divided into triangles)
64 Points, but only those used by the imported geometry

As when importing `*.neu` files, the wire radius must be set with the IP card preceding the IN card, and an ME card must be used when specifying dielectric surfaces in the same way as when the IN card is not present.

The user can also import points from the PATRAN file similar to importing points from FEMAP or NASTRAN files. The points defined in the PATRAN file will then available in PREFEKO as points (as if they were defined by DP cards) of the form `Txxx` where `xxx` is the index of the grid point. This may be used, for example, to attach additional structures to the geometry. In addition, the coordinate values of the point are available as variables in PREFEKO. For example, the variables `#t1234x`, `#t1234y` and `#t1234z` give the coordinates of the point with index 1234. Note that points are not included by default. Since points do not have an associated property ID, points are imported irrespective of their label.

8.2.17 IP Card



With this card a number of parameters, that define the density of the mesh, are set.

Parameters:

- R_1 Segment radius in m (it is scaled by the SF card).
- R_2 Maximum edge length of triangular elements in m (it is scaled by the SF card).
- R_3 Maximum segment length for wire segments in m (it is scaled by the SF card).
- R_4 Maximum edge length of dielectric cuboids in m (it is scaled by the SF card).
- R_5 Maximum edge length of tetrahedral volume elements (used with FEM) in m. (It is scaled by the SF card.)

The IP card only affects the commands following it, i.e. it has to be declared prior to the cards that define segments, triangles cuboids or tetrahedra.

It is possible to use more than one IP card in a file. This is necessary when a finer mesh is required in certain parts or when different radii occur in the geometry. For any command (e.g. the BL card) the previous IP card is applicable.

For the meshing process the following rules must be adhered to:

- The segment length l must be smaller than $\frac{\lambda}{10}$, where λ is the free space wavelength.
- The area A of each triangular element should be smaller than $\frac{\lambda^2}{70}$. For an equilateral triangle with side length s, the area is given by $A = \frac{\sqrt{3}}{4} s^2$. Using this formula, it is found that the edge length s should be shorter than approximately $\frac{\lambda}{5...6}$. According to the geometry and the need for accuracy, more or less triangles may be needed. If the memory constraints allow it, a segment length of $\frac{\lambda}{8...10}$ is preferred.
- When modelling a surface by means of a wire grid, the radius should be chosen so that the wire area in one direction is approximately the same as the area of the surface to be modelled as a wire grid. From the approximation $2\pi r \cdot l \approx l^2$, one finds the wire radius to be
$$r \approx \frac{l}{2\pi},$$
where $l \approx \frac{\lambda}{10}$ defines the segment length.

- The length on dielectric cuboids has to be small in comparison with the wavelength λ *in the dielectric* as well as the skin depth

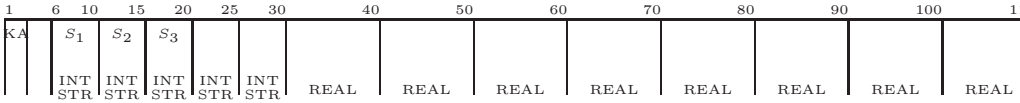
$$\delta = \sqrt{\frac{2}{\omega\mu\sigma}}.$$

- In some cases accurate modelling of the geometry requires significantly finer triangles and segments than specified by the guidelines above. (For low frequencies in particular, the segmentation rule of $\frac{\lambda}{10}$ is often much too coarse to yield a reasonable representation of the geometry.) Another case where finer discretisation may be required is where a wire runs parallel to a conducting plate. If the wire is closer than $\frac{\lambda}{10}$ to the plate, the size of the triangles in the direction orthogonal to the wire should be similar to the distance from the wire to the plate in order to give an accurate representation of the surface charge distribution.

If the segmentation rules are not adhered to, then the following errors and warnings will be reported:

	Warning	Error
Ratio of the segment length to the wavelength	$\frac{l}{\lambda} > 0.3$	$\frac{l}{\lambda} > 0.5$
Ratio of the segment radius to the segment length	$\frac{\rho}{l} > 0.3$	$\frac{\rho}{l} > 1.0$
Ratio of the area of the triangles to the wavelength squared	$\frac{A}{\lambda^2} > \frac{1}{30}$	$\frac{A}{\lambda^2} > \frac{1}{10}$
Ratio of the cuboid edge length to the wavelength	$\frac{l}{\lambda} > 0.25$	$\frac{l}{\lambda} > 0.5$
Ratio of the cuboid edge length to the skin depth	$\frac{l}{\delta} > \frac{1}{5}$	$\frac{l}{\delta} > \frac{1}{3}$

8.2.18 KA Card



With this card two points are joined, which form the border of the PO area. On this edge fringe wave currents are taken into account.

Parameters:

- S_1 Name of the begin point of the edge.
- S_2 Name of the end point of the edge.
- S_3 Label of the PO triangles adjacent to the PO border, i.e. the edge is assigned to all triangles with the same label.

The direction of an edge is arbitrary, i.e. it does not matter which edge point is chosen as the end or start point of the edge.

8.2.19 KK Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
KK	S_1	S_2	S_3	S_4			R_1	R_2	R_3	R_4	R_5			
	INT STR	INT STR	INT STR	INT STR	INT STR		REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card the shell of a cone or conical section may be created. (The cone can also be distorted to have an elliptical cross section.) A sketch is shown in figure 8-23.

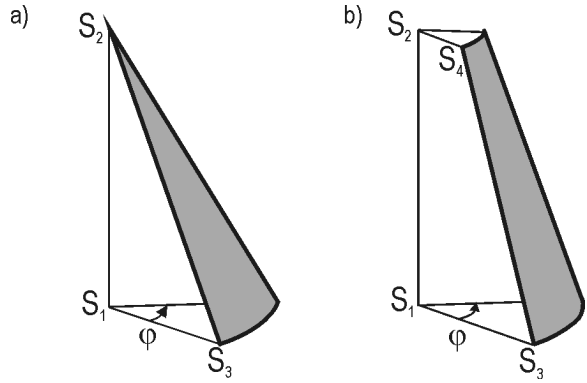


Figure 8-23: Sketch of the use of the KK card: a) cone and b) conical section.

Parameters:

- S_1 The name of a point on the axis of the cone in the plane of the base (or the larger circular side for a conical section).
- S_2 The name of a point indicating the vertex of the cone. This is the point in the plane of the smaller circle when creating a conical section.
- S_3 The name of a point on the radius of the base.
- S_4 If this parameter is defined, then a conical section is generated; if not then a cone is generated. S_4 is the point on the radius of the smaller circle and must be in the plane given by S_1, S_2 and S_3.
- R_1 The angle φ in degrees subtended by the conical section.
- R_2 Maximum edge length of the triangles along the arc of the cone, or the larger arc of the conical section in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.
- R_3 Only applies for conical sections: the maximum edges length of the triangles along the smaller of the arcs in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.

R_4 0: The normal vector of the generated triangles points to the outside.
1: The normal vector of the generated triangles points to the inside.

R_5 If this parameter is empty or is set to 1, a cone with a circular cross section is created. The parameter R_5 may be used to generate a cone with an elliptical cross section (within reasonable limits). $R_5 = \frac{b}{a}$ gives the ratio of the ellipse's two half axes, where a is the distance S_1–S_3. It is recommended to generate elliptical cones with extremely small or extremely large axial ratios with a CAD system (such as FEMAP) as the distortion formulation used in PREFEKO may fail in these cases.

The fineness of the mesh on the shell's surface is determined by the maximum edge length specified by the last IP card prior to the KK card. Along the arcs, accurate modelling of the geometry may require finer segmentation and the values R_2 and R_3 specify the maximum edge length along the corresponding arcs (R_3 is only used when a truncated cone is created). If either of these values is not specified the length specified with the IP card will be used on the corresponding arc.

First example of KK card usage:

Using the following command the conical shell in figure 8-24 is created.

```
**
IP                                          0.35
DP   A                          0.0         0.0         0.0
DP   B                          0.0         0.0         2.0
DP   C                          1.0         0.0         0.0
KK   A    B    C                360.0       0.3
EG
EN
```

Second example of KK card usage:

Using the following command the conical shell in figure 8-25 is created.

```
**
IP                                          0.35
DP   A                          0.0         0.0         0.0
DP   B                          0.0         0.0         0.5
DP   C                          1.0         0.0         0.0
DP   D                          0.5         0.0         0.5
KK   A    B    C    D           260.0       0.3         0.15
EG
EN
```

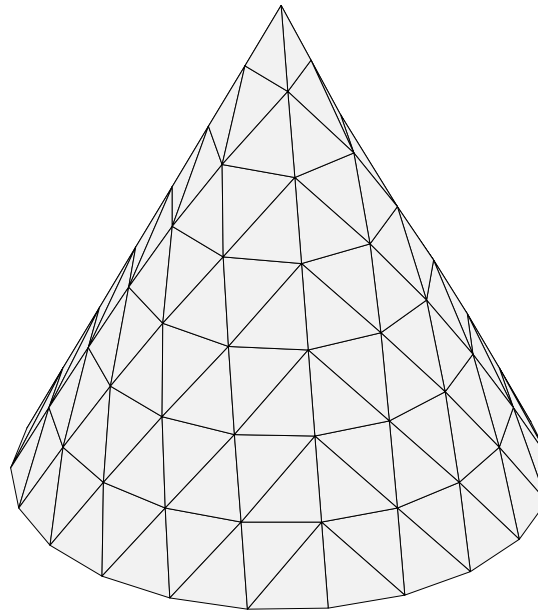


Figure 8-24: Example for the KK card

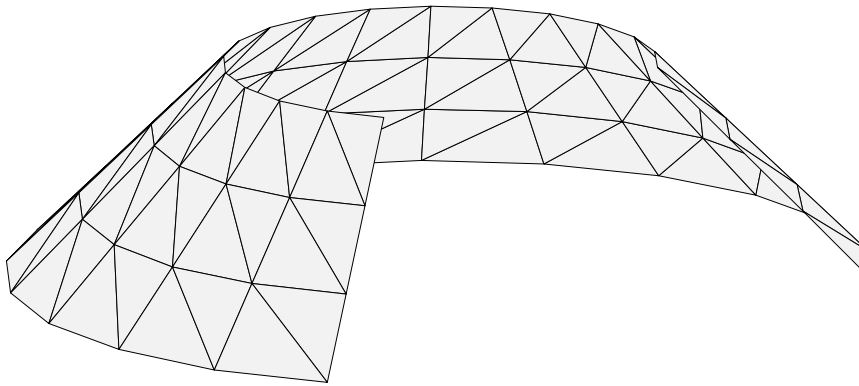


Figure 8-25: Second example for the KK card

Third example of KK card usage:

Oblique cones (e.g. the rear part of an aircraft fuselage) can also be produced. The commands below generates the conical shell shown in figure 8-26. For a truncated cone, the lines S_1–S_3 and S_2–S_4 must be parallel. The cross section is circular in planes that lies parallel to these lines and orthogonal to the plane containing all four points.

```
** oblique cone
IP                                        0.08
DP   A                        0           0          0
DP   B                        0.5         0          0.5
DP   C                        0.3         0          0
DP   D                        0.6         0          0.5
KK   A    B    C    D         360         0.08       0.08       0

EG
EN
```

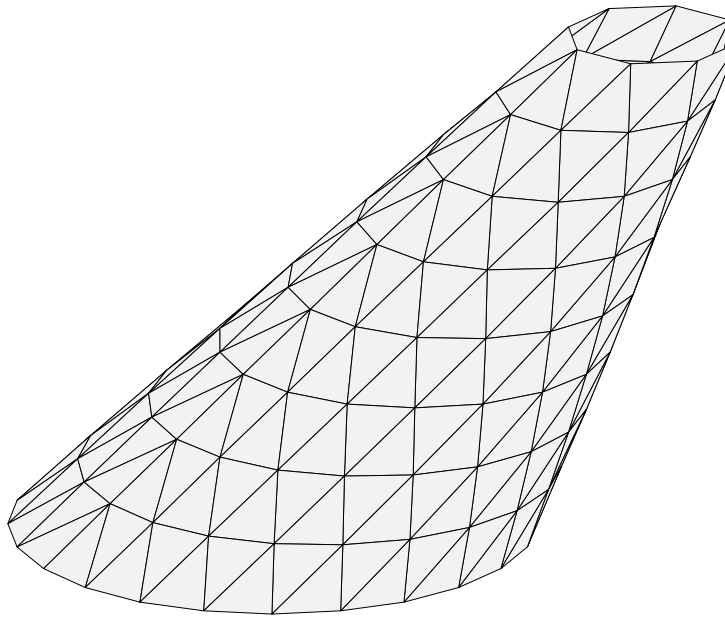


Figure 8-26: Oblique example for the KK card

Fourth example of KK card usage:

A cone section with an elliptical cross section, as shown in figure 8-27, is generated with the following commands.

```
** example for an elliptical cone
#a   = 1.8    ** first half axis
#b   = 1.0    ** second half axis
#hct = 3.5    ** height of the cone
#rct = #a/2   ** radius of the whole at the top cylinder
#offs = #a/2  ** offset for the cylinder

** segmentation
#edge_l = 0.4
IP                                      #edge_l

** define points
DP   B                          0          0          0
DP   D                          #a         0          0
DP   E                          #rct+#offs0           #hct
DP   F                          #offs      0          #hct

** define the geometry
KK   B    F    D    E          360        #edge_l    #edge_l    0          #b/#a

EG
EN
```

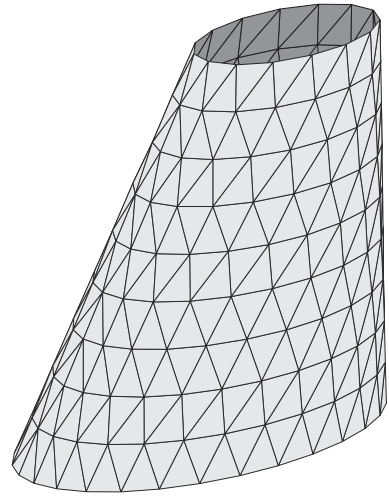


Figure 8-27: Example of a cone with an elliptical cross section

8.2.20 KL Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
KL	K_1	K_2	P_0	P_N		L_0	L_N							
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

With this card a wedge in the PO area is defined, on which correction terms for the surface current density are defined on both sides of the wedge. Figure 8-28 shows a sketch.

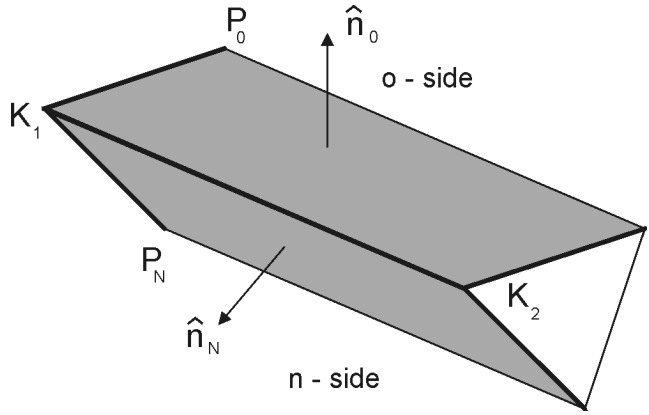


Figure 8-28: Sketch illustrating the use of the KL card

Parameters:

K_1	The name of the begin point of the edge of the wedge.
K_2	The name of the end point of the edge of the wedge.
P_0	A point on the o side of the wedge.
P_N	A point on the n side of the wedge.
L_0	The label of the PO triangles that are adjacent to the wedge on the o side. This means that the corresponding correction term for the o side is assigned to the PO triangles that have this label.
L_N	The label of the PO triangles that are adjacent to the wedge on the n side. This means that the corresponding correction term for the n side is assigned to the PO triangles that have this label.

8.2.21 KR Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
KR	S_1	S_2	S_3	S_4			R_1	R_2	R_3	R_4				
	INT STR	INT STR	INT STR	INT STR	INT STR		REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card a circular region with or without a hole is created. (It is also possible to distort it to an elliptical region.) A sketch is shown in figure 8-29.

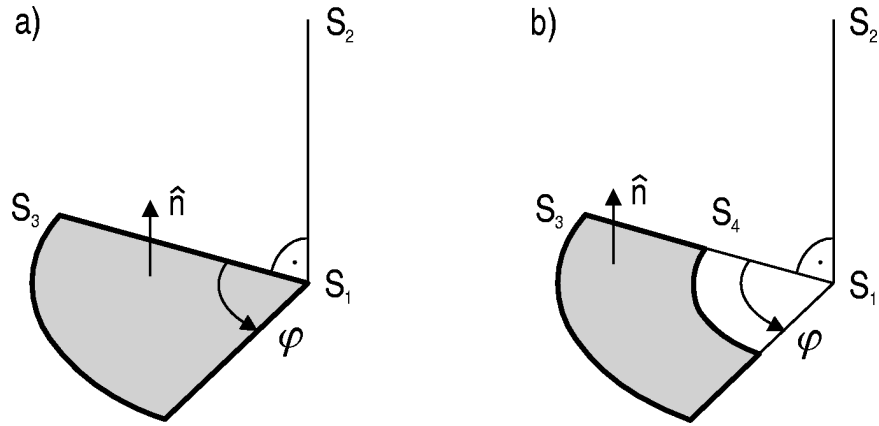


Figure 8-29: Sketch illustrating the use of the KR card

Parameters:

- S_1 Name of the centre point of the circle.
- S_2 The name of a point that is situated at any distance perpendicular to and above the plane of the circle.
- S_3 The name of the point where the arc of the circular segment begins.
- S_4 If there is a value present for this parameter, then a circular ring is created. S_4 is the inner corner point. S_4 must lie between S_1 and S_3.
- R_1 The angle φ in degrees subtended by the arc.
- R_2 The maximum edge length of the triangles along the outer edge of the arc in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.
- R_3 When a disk with a hole is created, the maximum edge length for the triangles along the inner edge of the arc in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.

R_4 If this parameter is empty or is set to 1, a circular disk is created. Within certain limits, the parameter R_4 may be used to generate an elliptical disk. $R_4 = \frac{b}{a}$ gives the ratio of the two half axes, where a is the distance S_1–S_3. It is recommended to generate ellipses with extremely small or extremely large axial ratios with a CAD system (such as FEMAP) as the distortion formulation used in PREFEKO may fail in these cases.

The circle's plane is perpendicular to the line S_1–S_2. This length is arbitrary. The radius of the disc is given by the length between the points S_3 and S_1. The area that is to be subdivided (the shaded region in figure 8-29) is generated by sweeping the edge S_3–S_1 around the axis S_1–S_2 through φ degrees in the mathematically positive sense. For $\varphi = 360°$ a circle is obtained.

The fineness of the mesh is determined by the maximum edge length specified by the last IP card prior to the KR card. Along the arcs, accurate modelling of the geometry may require finer segmentation and the values R_2 and R_3 specify the maximum edge length along the outer and inner (if applicable) arcs respectively. If either of these values is not specified the length specified with the IP card will be used on the corresponding arc.

The normal vector of the triangles on the disk all point in the direction from S_1 to S_2.

First example of KR card usage:

Using the following commands the circular disc, shown in figure 8-30, is created.

```
**
IP                                          0.4
DP   A                          0.0         0.0         0.0
DP   B                          0.0         0.0         1.0
DP   C                          1.0         0.0         0.0
KR   A    B    C                360.0       0.35
EG
EN
```

Second example of KR card usage:

Using the following commands the circular disc, shown in figure 8-31, is created.

```
**
IP                                          0.3
DP   A                          0.0         0.0         0.0
DP   B                          0.0         0.0         1.0
DP   C                          1.0         0.0         0.0
DP   D                          0.3         0.0         0.0
KR   A    B    C    D          180.0       0.3         0.15
EG
EN
```

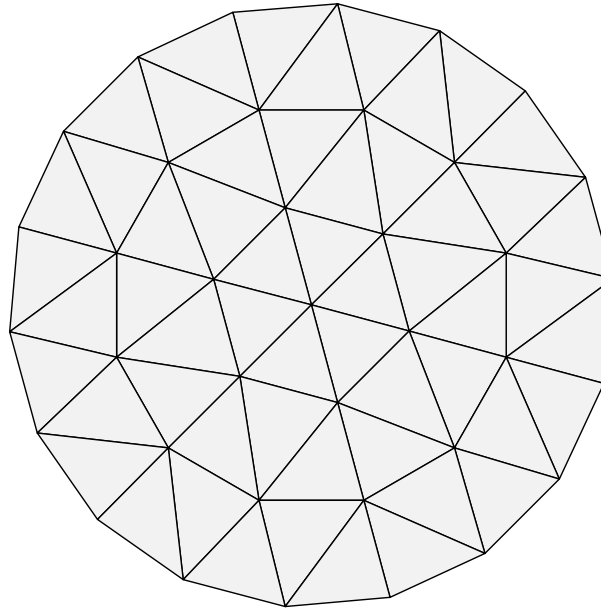


Figure 8-30: First example for the KR card

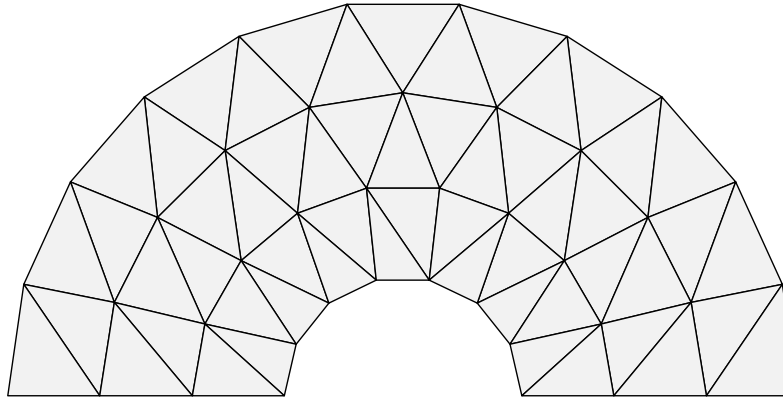


Figure 8-31: Second example for the KR card

Third example of KR card usage:

The elliptical ring disk, shown in figure 8-32, is created with the following commands

```
** Example for an elliptical plate with hole

** general parameters
#a   = 2.5    ** first half axis of the ellipse
#b   = 0.9    ** second half axis of the ellipse
#c   = 1.7    ** radius (first half axis) of the hole

** segmentation
#edge_l = 0.35
IP                                      #edge_l

** define points
DP   A                          0         0         0
DP   B                          0         0         #a
DP   C                          #a        0         0
DP   D                          #c        0         0

** define the geometry
KR   A    B    C    D          90        #edge_l   #edge_l   #b/#a
SY   1    1    1    0

** end
EG
EN
```

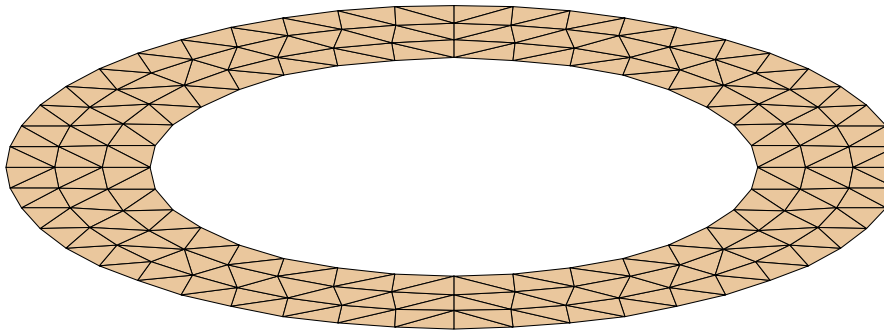


Figure 8-32: Example of an elliptical disk with a hole created by setting R_4

8.2.22 KU Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
KU		S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4	R_5			
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card a spherical segment can be generated. Figure 8-33 shows an example.

Parameters:

- S_1 Name of the centre of the sphere.
- S_2 The name of a point that indicates the $\vartheta = 0$ direction in a spherical coordinate system. The distance between S_1 and S_2 is the radius of the sphere.
- S_3 The name of a point that indicates the $\varphi = 0$ direction in a spherical coordinate system. The distance between S_1 and S_3 is the radius of the sphere.
- S_4 0: The normal vector of the generated triangles point to the outside, i.e. away from the centre of the sphere.
  1: The normal vector point to the inside (centre) of the sphere.
- R_1 The begin angle ϑ_a in degrees of the spherical segment.
- R_2 The begin angle φ_a in degrees of the spherical segment.
- R_3 The end angle ϑ_e in degrees of the spherical segment.
- R_4 The end angle φ_e in degrees of the spherical segment.
- R_5 The maximum length of the triangles along the curved edges in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.

A complete sphere may be created with the parameters $\vartheta_a = \varphi_a = 0$, $\vartheta_e = 180°$ and $\varphi_e = 360°$.

Example of KU card usage:

Using the following commands the spherical segment in figure 8-34 is generated.

```
**
IP                                        0.25
DP   A                          0.0       0.0       0.0
DP   B                          0.0       0.0       1.0
DP   C                          1.0       0.0       0.0
KU   A    B    C                60.0      90.0      180.0     270.0     0.25
EG
EN
```

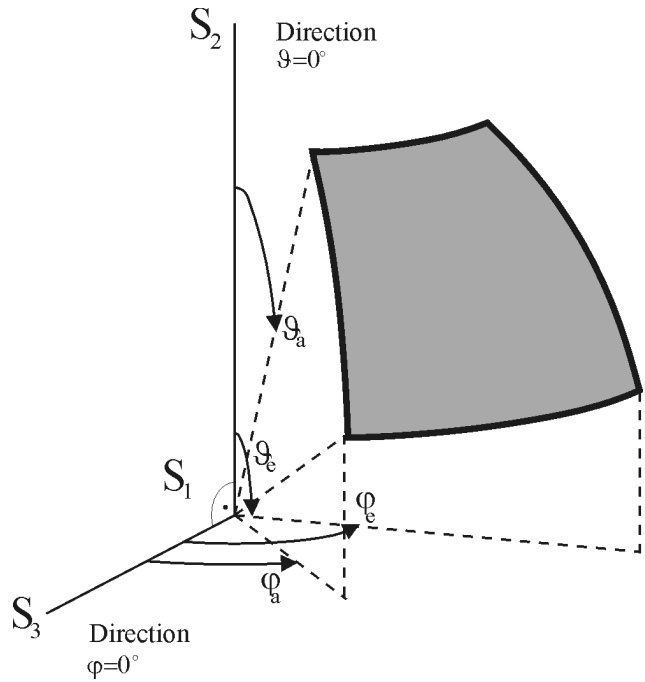


Figure 8-33: Sketch illustrating the use of the KU card

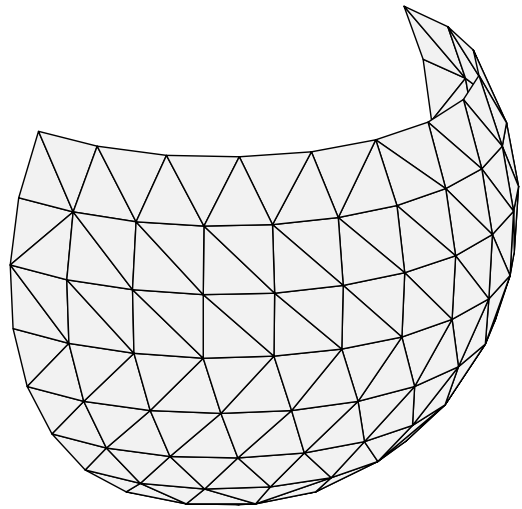
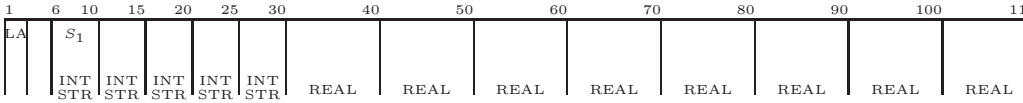


Figure 8-34: Example for the KU card

8.2.23 LA Card



With this card, labels are assigned to segments, triangles, polygons, cuboids, UTD cylinders and points.

Parameters:

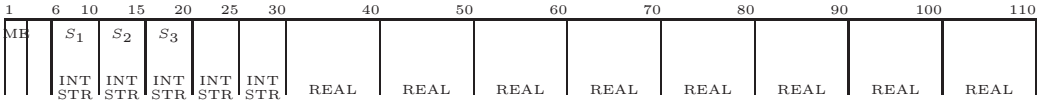
S_1 The label assigned to all segments, triangles, etc. defined in cards following this one (a positive number in the range 0 to 99999).

In order to select the position of the feed (Ax cards)[8], the location of impedance loading (LD, LS, LP and LZ cards) or structures on which to apply the skin effect (SK cards), each segment, triangle, etc. is assigned a label. Other applications include the selective transformation or copying of parts of geometry (TG card), and outputting of currents on selected elements (OS card). Labels are also used to define triangles on which to apply physical optics (PO card).

All elements, etc. that are created after the LA card (e.g. by a BP card), are assigned the value S_1 as label. A different label is only assigned by a new LA card. All structures created before the first LA card (or if no LA card is present), is assigned the default label which is 0.

[8]The definition Ax stands for any of the control cards A0, A1, A2, ...

8.2.24 ME Card



When solving the fields in dielectric objects by means of the surface current method, this card can be used to distinguish the different media or to create segments and metallic triangles within the dielectric. Furthermore, this card is used to switch between the generation of metallic triangles and triangles that represent the surface of the dielectric. Another special case is when metallic triangles represent the surface of a dielectric object (e.g. a dielectric that has been coated with metal).

All the segments that follow this card are assigned the properties of the medium in which they are found. Triangles are treated differently — it depends upon whether they are metallic triangles or triangles on the boundary of a dielectric object. Here the properties of the media are assigned to the respective sides.

Parameters:
There are a number of different cases:

- Fields S_2 and S_3 empty

  S_1 All the geometry cards generating segments and triangles that follow this card, are assigned the medium properties with the index S_1. $S_1 = 0$ is for the special case, where the triangles and segments are situated in free space.

- Fields S_2 occupied and S_3 empty

  S_1, S_2 All segments generated after this card are situated in the medium S_1. All the triangles represent a boundary between dielectric objects. The normal vector points from medium S_1 to medium S_2. (Note that the user should ensure that the normal vectors of the triangles are in this direction. It is recommended to use WinFEKO to validate the geometry.)

- Field S_3 occupied

  S_1, S_2 All triangles generated after this card are metallic triangles that are situated on the surface of a dielectric object between the media S_1 and S_2. The normal vector points from medium S_1 to medium S_2.

  S_3 No special meaning, input field must just be occupied and not empty.

All triangles and segments before a ME card represent metallic structures in free space. This is also the case when an input file does not have a ME card.

When, for example, the surface current method is used to generate a dielectric sphere (which shall represent, in this example, medium 1) in free space, the following card form is used:

```
 ...

DP   A                          0.0       0.0       0.0
DP   B                          0.0       0.0       1.0
DP   C                          1.0       0.0       0.0
ME   1    0
KU   A    B    C    0          0.0       0.0       180.0     360.0

 ...
```

The normal vectors of the sphere point outwards (see KU card, section 8.2.22) from medium 1 (dielectric) to medium 0 (free space).

More detail can be found in `example_04` and `example_23` (*Examples Guide*). As an illustration, consider the basic geometry of `example_23` as shown in figure 8-35 and consists of a dielectric cone (medium 1) mounted on top of a metallic cylinder.

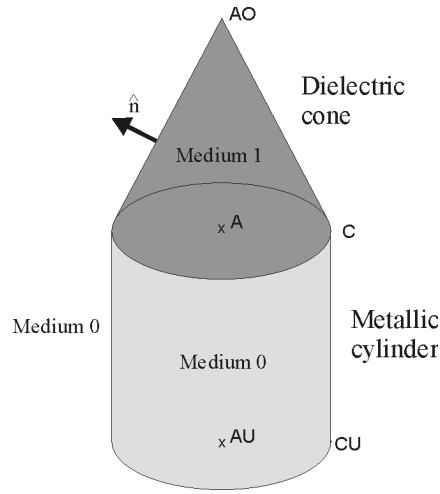


Figure 8-35: Example of a dielectric cone on top of a metallic cylinder, to demonstrate the use of the ME card.

There are three types of triangles involved:

- Metallic triangles in free space (Medium 0) on the bottom and side of the cylinder
- Metallic triangles also forming the border surface of the dielectric body on the lid of the cylinder (which is also the basis of the dielectric cone)
- Dielectric triangles forming the surface of the dielectric body (the boundary between medium 1 — the inner dielectric region — and medium 0 — the free space outer region) on the top surface of the cone

Since there is symmetry in two planes, only a quarter of the structure is generated. The geometry part of the input file is given below; detailed comments being included with the ME card.

```
** Definition of the variables and discretisation parameters
#lambda = 1.0       ** wavelength
#a = 0.3*#lambda    ** radius of cylinder
#h = 0.6*#lambda    ** height of cylinder and cone
#epsr = 2           ** relative dielectric constant
#side_l = #lambda / sqrt(#epsr) / 6
IP                                      #side_l

** Definition of the points
DP   A                          0         0         0
DP   AO                         0         0         #h
DP   AU                         0         0        -#h
DP   C                          #a        0         0
DP   CU                         #a        0        -#h

** Generate a quarter of the geometry
**   First generate the metallic surfaces that are found in
**   free space (medium 0)
**   Either leave out the ME card or include it like this:
ME   0

**     Basis of the cylinders
KR   AU   A    CU              90        #side_l
**     Surface of the cylinders
ZY   AU   A    CU              90        #side_l
```

```
**   Now the outer surface of the cone is generated as the outer
**   surface of a dielectric body, where the normal vector points
**   from medium 1 to medium 0 (as shown in the drawing)
ME   1    0
KK   A    AO   C               90        #side_l   #side_l   0

**   Finally generate the metallic surfaces on the dielectric boundary
**   (metallic triangles on the surface of a dielectric body, normal
**   vector pointing from medium 0 to medium 1)
ME   0    1    1
KR   A    AO   C               90        #side_l

** Introduction of symmetry
SY   1    2    3

** End of geometry section
EG   1

** Specify the dielectric parameters for medium 1 (medium 0 defaults
** to free-space, no need to set the parameters separately)
DI   1     ...

** Additional cards (excitation, frequency, calculation requests)
...

** End
EN
```

8.2.25 NU Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
NU		I_1 S_1 INT STR	I_2 S_2 INT STR	S_3 INT STR	S_4 INT STR	S_5 INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card generates and meshes a NURBS surface from the specified control points.

Parameters:

- I_1 The order p of the Bézier curve in the $\hat{u}$-direction. The range is $1 \leq p \leq 4$, where $p = 1$ is linear, 2 quadratic, and so on.
- I_2 The order q of the Bézier curve in the $\hat{v}$-direction, $1 \leq q \leq 4$.

This line is followed by p+1 lines with the node names of q+1 control points in the fields S_1 to S_{q+1} (as many of the 5-column fields as required). A surface that is linear in the $\hat{u}$-direction and quadratic in the $\hat{v}$-direction would, for example, require three lines.

```
NU   1    2
     AA   AB   AC
     BA   BB   BC
```

It is possible to create a triangular Nurbs surface. In this case all control points on one side must be identical (use the same point). The weights of the control points are specified at the DP card. Note that, for higher order Bézier curves, the surface does not pass through the control points except those on the corners.

First example of NU card usage:

The "saddle point" shape in figure 8-36 is generated with

```
** Demo for the NU card

DP   AA                          0          0          0
DP   AB                          0          1          0.5
DP   BA                          1          0.2        0.6
DP   BB                          1.2        0.9        -0.1

IP                                          0.1
NU   1    1
     AA   AB
     BA   BB

EG
EN
```

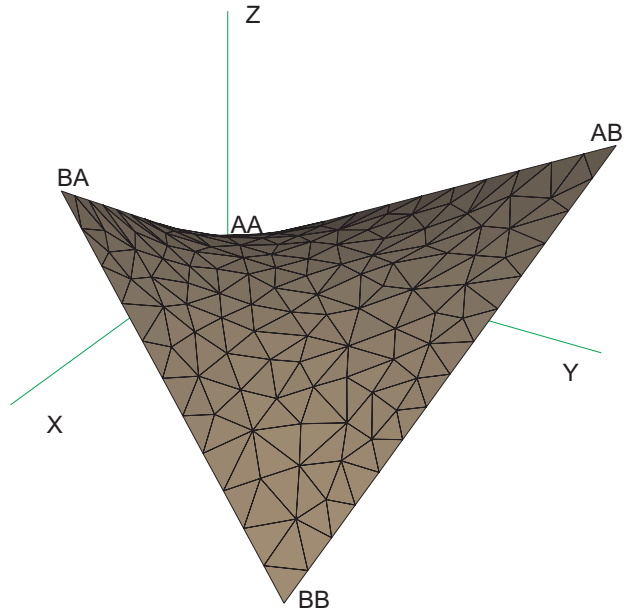


Figure 8-36: "Saddle point" example using the NU card

Second example of NU card usage:

NURBS may also be used to generate flat surfaces with curved edges. The section of a circular plate with a square hole in figure 8-37 is generated with the file

```
** Demo for the NU card
** Square hole in a circular plate

** Structure variables
#r = 1.0                  ** Radius of circle
#a = 0.4                  ** Half width of rectangle
#edg_len = 0.15           ** Maximum edge length

** We will use a first order interpolation in the radial direction and
** second order along the arc (thus also along the border of the hole)

** Data points along the hole (just a straight line)
DP   P0                       #a        0         0         1
DP   P1                       #a        #a/2      0         1
DP   P2                       #a        #a        0         1
```

```
** Data points along the 45 degree arc
** The first two control points lies at the start (Q0) and end (Q2)
** points of the arc.  The third control point (Q1) lies at the cross
** point of the two lines that lie tangential to the arc at Q0 and Q2
** respectively.  Define the y value of this point.
#y_Q1 = #r * (sqrt(2) - 1)

** With M lying on the middle of the line between Q0 and Q2 and S on the
** intersection between the circular arc and the line between M and Q1,
** the weight of point Q1 is the ratio of the lengths M-S to S-Q1
#w_Q1 = sqrt(2 + sqrt(2))/2
DP   Q0                      #r          0.0         0.0         1
DP   Q1                      #r          #y_Q1       0.0         #w_Q1
DP   Q2                      #r/sqrt(2)#r/sqrt(2)0.0         1

** Create the Bezier surface
IP                                       #edg_len
NU   1    2
     Q0   Q1   Q2
     P0   P1   P2

** Copy and rotate the 45 degree structure to a quarter disk
TG   1    0    0         0     180.0              -90.0

EG
EN
```

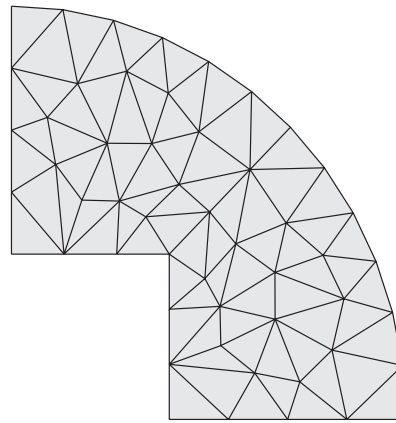


Figure 8-37: Demonstation of using the NU card to generate flat surfaces

Third example of NU card usage:

The linear-quadratic shape in figure 8-38 is generated with

```
** Linear-quadratic demo for the NU card

IP                                      0.1

DP   AA                      0.1        0          0
DP   AB                      0          1          0.3
DP   AC                      0.1        2          0.1        0.8
DP   BA                      1          0.2        0.3        0.6
DP   BB                      1.2        0.9        -0.2
DP   BC                      1.1        1.8        0.1        1.2

NU   1    2
     AA   AB   AC
     BA   BB   BC

EG
EN
```

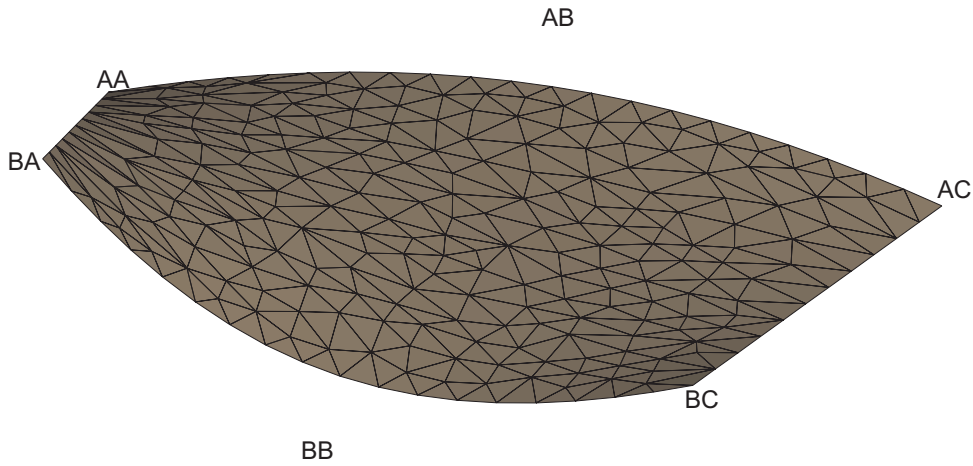


Figure 8-38: Linear-quadratic example using the NU card

8.2.26 PB Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
PB		S_1	S_2	S_3	S_4		R_1	R_2						
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card can be used to generate a part of a parabolic reflector as shown in figure 8-39.

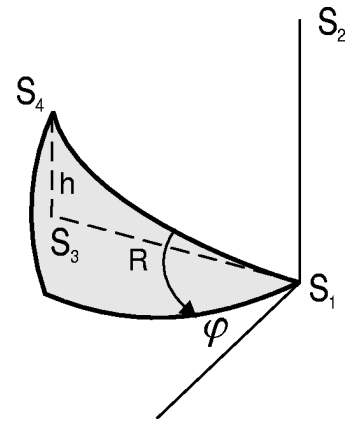


Figure 8-39: Sketch illustrating the use of the PB card

Parameters:

- S_1 Name of the centre of the paraboloid.
- S_2 Name of a point at any distance and perpendicular to the base plane and above the centre point.
- S_3 Name of a point, on the outer edge of the paraboloid, but on the base plane.
- S_4 Name of a point, in the direction S_1–S_2 with respect to point S_3 and on the edge of the paraboloid.
- R_1 The angle φ in degrees subtended by the arc of the parabolic reflector.
- R_2 Maximal edge length of the triangles along the outer edge of the arc in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.

The radius R of the paraboloid is derived from the distance between the points S_1 and S_3, as can be seen in figure 8-39. The height is determined by the distance between points S_1 and S_3. The focal point f is determined by:

$$f = \frac{R^2}{4h}.$$

Example of PB card usage:

The parabolic reflector as shown in figure 8-40 can be generated by using the following lines

```
IP                                          0.2
DP   A                          0           0          0
DP   B                          1           0          0
DP   C                          0           0          1
DP   D                          0.5         0          1
PB   A    B    C    D          360         0.2
EG
EN
```

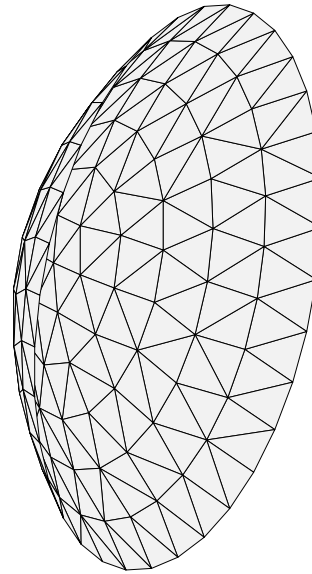


Figure 8-40: Example for the PB card

8.2.27 PH Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
PH	S_1	S_2	S_3	S_4				R_2		R_4				
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

This cards creates a triangular or quadrangular plate with a circular or elliptical hole as shown in figure 8-41. The hole can be used, for example, to attach a cylinder (ZY card) to the plate and it can be filled with the KR card.

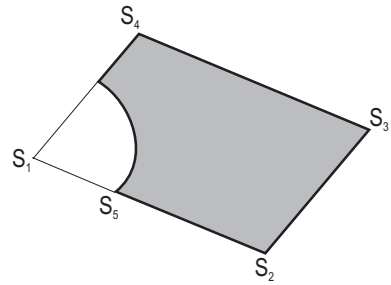


Figure 8-41: Sketch illustrating the use of the PH card

Parameters:

- S_1 Name of the corner point where the hole is located (it is also the centre of the hole).
- S_2 Name of the second corner of the plate.
- S_3 Name of the third corner of the plate. If this field is left empty, a triangular plate is created.
- S_4 Name of the fourth corner of the plate.
- S_5 Name of a point on the line S_1–S_2 that forms the starting point of the circle or ellipse bordering the hole. The special case where S_5 is identical to S_2 is supported, but due to the resulting geometry yields triangles with very small angles.
- R_2 The maximum edge length of the triangles along the edge of the hole in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.
- R_4 If this parameter is empty or is set to 1, a circular hole is created. Within certain limits, the parameter R_4 may be used to generate an elliptical hole. $R_4 = \frac{b}{a}$ (where a is the distance S_1–S_5) then gives the ratio of the two half axes of the ellipse.

First example of PH card usage:

The commands below create the rectangular plate shown in figure 8-42. Note the extremely narrow triangles at the corners as mentioned above.

```
** Example for the PH card

** Size of the plate
#a = 1
#b = 1
** Size of the circular hole
#hole = 1

** Discretisation
#edge_l  = #a / 10
IP                                      #edge_l

** Define the points
DP   A                          0          0          0
DP   B                          #a         0          0
DP   C                          #a         #b         0
DP   D                          0          #b         0
DP   E                          #hole      0          0

** The plate with hole
PH   A    B    C    D    E               #edge_l

EG
EN
```

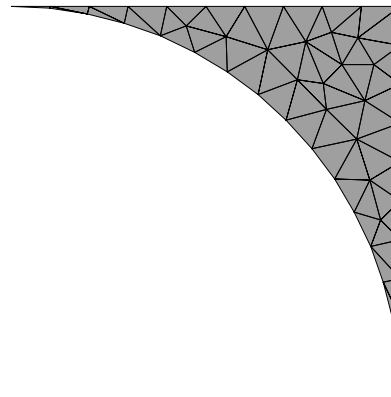


Figure 8-42: Example using the PH card

Second example of PH card usage:

Figure 8-43 shows a rectangular plate attached to an elliptical cylinder created with these commands.

```
** Example of attaching the PH card to a cylinder

** Size of the plate
#a = 1.2
#b = 1.7

** Size of the elliptical cylinder
#ella = 0.7
#ellb = 0.3
#h = 0.3
#ratio = #ellb / #ella

** Discretisation
#edge_l  = #a / 10
#edge_ld = 0.8 * #edge_l
IP                                      #edge_l

** Define the points
DP   A                        0         0         0
DP   B                        #a        0         0
DP   C                        #a        #b        0
DP   D                        0         #b        0
DP   E                        #ella     0         0
DP   F                        0         0         #h

** The plate with hole
PH   A    B    C    D    E              #edge_ld            #ratio

** The cylinder
ZY   A    F    E    0         90        #edge_ld  #ratio

EG
EN
```

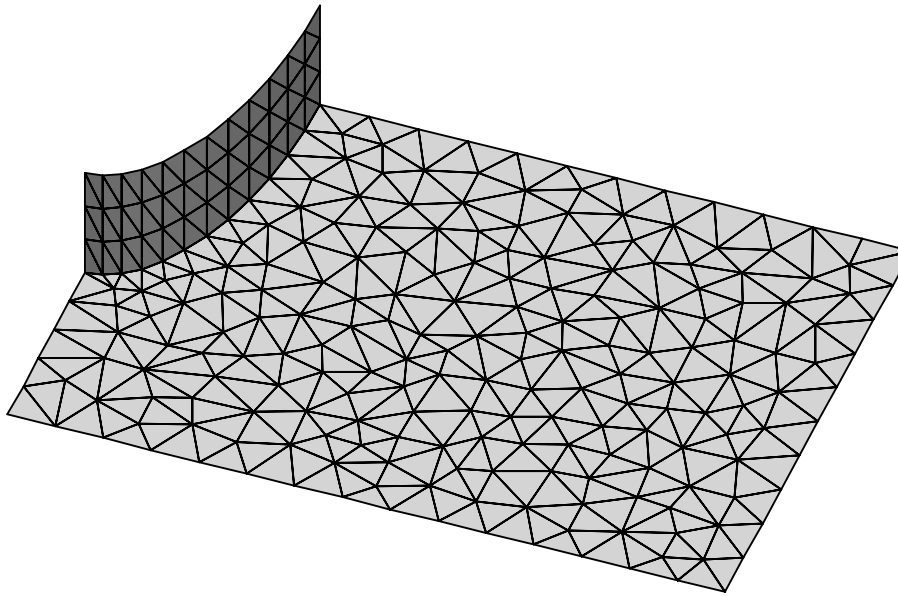


Figure 8-43: Example of attaching a plate (PH card) and cylinder
(second example of PH card usage)

Third example of PH card usage: The quadrangle shown in figure 8-44 is created with the lines

```
** discretisation
IP                                    0.1

** define some points
DP   A                        0       0       0
DP   B                        1       0       0
DP   C                        0.8     0.7     0
DP   D                        0.2     0.6     0
DP   E                        0.4     0       0

** the plate with hole
PH   A    B    C    D    E                                        0.6

EG
EN
```

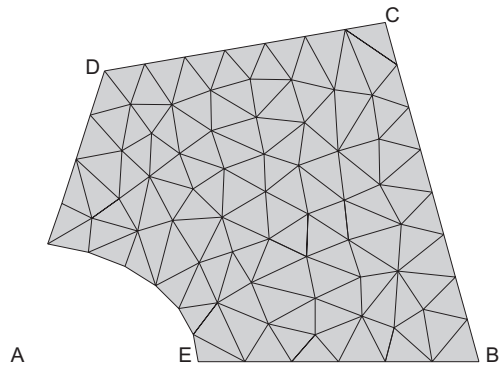


Figure 8-44: Example of a quadrangular plate with an elliptical hole

Fourth example of PH card usage: If the point D above (corresponding to S_3) is not used, the triangle in figure 8-45 is created (using 0.1 m maximum edge length).

```
** define some points
DP   A                        0         0         0
DP   B                        1         0         0
DP   D                        0.2       0.6       0
DP   E                        0.4       0         0

** the plate with hole
PH   A    B         D    E

EG
EN
```

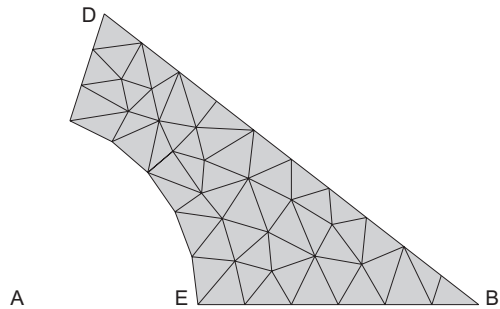


Figure 8-45: Example of a triangular plate with a spherical hole

8.2.28 PM Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
PM	AM_A	BM_B	CM_C	DM_D	I_5	EM_E	FM_F	GM_G	...					
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

This card defines (by specifying the corner points) a polygonal plate which must be meshed into triangles. Concave corners are allowed. A sketch is shown in figure 8-46. The user can also specify a smaller or larger mesh size along certain edges.

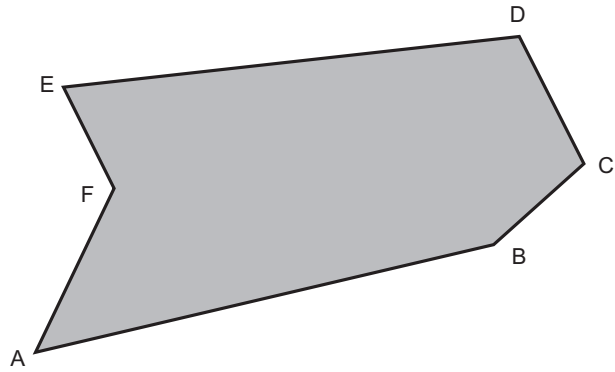


Figure 8-46: Sketch illustrating the use of the PM card

A maximum of 13 corner points are allowed. The points are connected in the order in which they are entered in the PM card. Concave corners are allowed. The corner points have to be defined by a DP card prior to the PM card. The point names are specified in the first four string parameters (columns 6–10, 11–15, 16–20 and 21–25) and the first nine real parameters (columns 31–40, 41–50, ..., 111–120). The point names may not be longer than 5 characters, but may be aligned at any position in the 10-character real fields.

If non-uniform meshing is required, the mesh size along each edge is specified in the next 13 real parameters (columns 121–130, 131–140, ..., 241–250), where the first mesh size refers to the edge from the first to the second point, the next to the edge between the second and third point, etc. There may only be as many mesh sizes as node points and the last size refers to the edge from the last listed node point to the first point. If any edge size is left empty, the value specified with the IP card will be used.

The user can also specify internal mesh points, for example, to create connection points. If the parameter I_5 is set to 0 no internal mesh points are specified. If $I_5 = 1$ a second line, which specifies up to 26 internal mesh points (specified with DP cards prior to the PM card), must follow the PM card. The point names are specified similar to the points in the first line except that there are 22 real fields instead of just 9.

Example of PM card usage:

The commands below generate a plate with a concave corner (which one might also define with two BQ cards) as shown in figure 8-47 — note the finer mesh along the edges from B to C and C to D.

```
** Example for the PM card (meshed polygon)

DP   A                          0.0       1.0
DP   B                          0.0       0.5
DP   C                          1.0       0.5
DP   D                          1.0       0.0
DP   E                          2.0       0.0
DP   F                          2.0       1.0

IP                                       0.2
PM   A    B    C    D         E         F      ...
                                                    0.1       0.1

EG
EN
```

The `...` above indicates that everything should be on a single line with the first `0.1` starting in column 131.

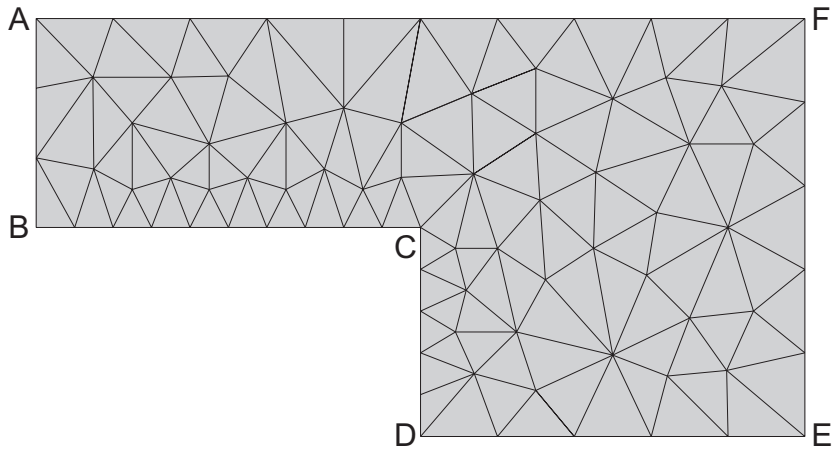


Figure 8-47: Example for the PM card

Second example of PM card usage:

The commands below generate a plate with three specified internal mesh points as shown in figure 8-48. Note that there are node points at Q_1, Q_2 and Q_3.

```
** Example for the PM card (meshed polygon) with internal points

**   External boundary points of the polygonal plate
DP   P1                        0.1       0.1       0
DP   P2                        1.2       0         0
DP   P3                        1.5       1.5       0
DP   P4                        0.3       1.2       0
DP   P5                        -0.3      0.5       0

**   Internal mesh points
DP   Q1                        0.5       0.5       0
DP   Q2                        1.0       0.5       0
DP   Q3                        0.7       1.0       0

**   Mesh the plate
IP                                       0.2
PM   P1   P2   P3   P4   1    P5
     Q1   Q2   Q3

EG
EN
```

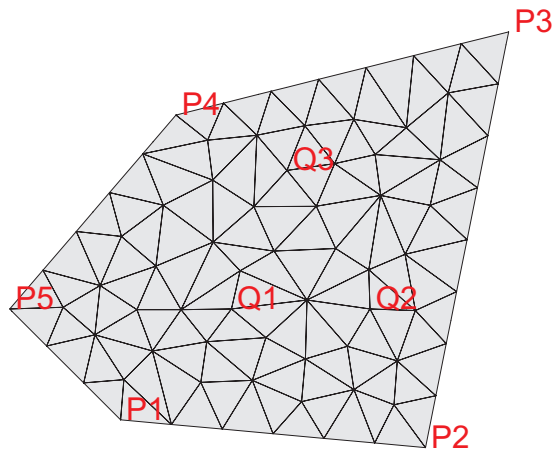


Figure 8-48: Example for the PM card with internal mesh points

8.2.29 PO Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
PO		PO LAB	NOSH ADE	PO SYM FLAG	PO NOCO UPL			PO MAX REFL	R_3	R_4	SAVE VIS			
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card the application of the physical optics approximation is possible.

Parameters:

POLAB All metallic/dielectric triangles and polygons, that have the label POLAB, are treated with the physical optics approximation.

NOSHADE 0: Normal, ray tracing is carried out.

1: The ray tracing is switched off to save computational time. The assumption is made that all triangles on which the PO approximation is made are illuminated by the source and the moment method area. The side in relation to the normal vector is automatically determined.

2: Full ray tracing is done as for NOSHADE=0, but metallic triangles can only be lit from the side to which the normal vector is pointing. This has two applications:

- Acceleration of the PO ray tracing with closed bodies (the normal vector must then point outward), since the dot product of the normal and propagation vectors can be used to quickly determine if a triangle or polygon is to be used in the ray tracing. In this case the closed model must be constructed with the normals pointing outward.
- In, for example the MoM/PO hybrid method on a closed body, the MoM region (such as an antenna) can be prevented from illuminating the PO region from inside.

POSYMFLAG If ray tracing is done with NOSHADE=0, then symmetry is used. Thus the computational time is reduced, when determining the shading. For electric and magnetic walls, it is assumed that symmetrical shading is possible. If geometrical symmetry is used, then for POSYMFLAG=0 full ray tracing is done, but for POSYMFLAG=1 symmetry is utilised. It is possible to e.g. define half a plate and create the other half through symmetry. An asymmetric object may then placed in front of the plate. In this case only POSYMFLAG=0 will function correctly.

PONOCOUPL
- 0: usual option, coupling between MoM and PO *is* considered.
- 1: The coupling between the MoM region and the PO region is neglected. The implication is that the currents in the PO region has no effect on the current distribution in the MoM region. This option, which should lead to some saving in computational time and storage space, is especially useful when the PO region and the MoM is not directly adjacent.

POMAXREFL This parameter determines the number of reflections to be taken into account for triangles with labels in the specified range. (For example, POMAXREFL must be at least 2 to calculate the scattering from a dihedral and at least 3 for a trihedral.)

R_3 If R_3 is empty the PO is applied to all triangles with label POLAB. If R_3 is set, the PO is applied to all triangles with labels in the range from POLAB to R_3 (both end values included).

R_4 The ray tracing required for the physical optics is accelerated by recursively subdividing the problem domain, a so called "multilevel tree". It must balance memory requirement against speedup — both increase as the number of levels increases. The number of levels is determined by specifying the number of triangles at the lowest level:

- -1: The "multilevel tree" is not used.
- 0: The number of triangles is automatically selected by the program and should be approximately optimal for most problems.
- else: The value of R_4 specifies the number of triangles at the lowest level. It should be larger than 2 and at least a factor of 10 less than the number of triangles in the problem.

SAVEVIS
- 0: Normal execution.
- 1: The PO visibility information is stored in a `*.vis` file for later reuse.
- 2: The PO visibility information is read from the `*.vis` file, i.e. the calculation of the visibility information is skipped. For large models this can result in considerable time saving.
- 3: If a `*.vis` file exists, the PO visibility information is read from this file. Otherwise the information is calculated and saved in a `*.vis` file for later use.

The physical optics (PO) approximation can only be used for certain structures. Structures where the antenna is situated in front of a reflector are well suited. Then PO can be used for the triangles that form the reflector. This results in a large reduction in computational time and memory for electrically large objects.

If triangles/polygons with different labels are to be approximated with PO, then more than one PO card may be used. The parameters NOSHADE and POMAXREFL can be specified for each label, but the parameters POSYMFLAG, PONOCOUPL, R_4 and SAVEVIS are global. For the global parameters, the values of the last PO card will be used.

A basis function that has been assigned to an edge between two triangles will only be solved with the PO, if the PO approximation has been declared for the labels of both triangles.

The metallic PO region must be perfectly conducting, i.e. no losses are allowed.

8.2.30 PY Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
PY		*A*	*B*	*C*	*D*		*E*	*F*	*G*	...				
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card defines (by specifying the corner points) a polygonal surface to which the UTD or PO formulation is applied. The UTD is applied unless the PO is specified for the label of the polygon. A sketch is shown in figure 8-49.

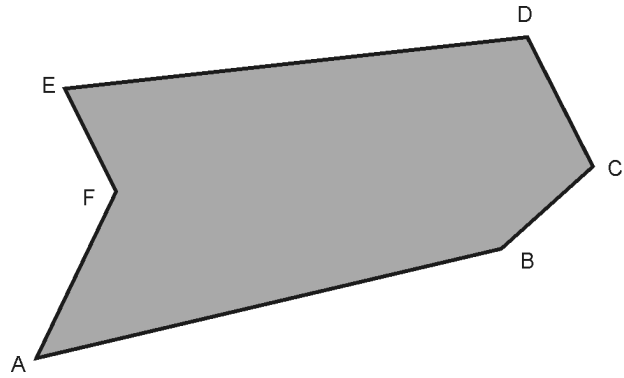


Figure 8-49: Sketch illustrating the use of the PY card

Parameters:

A Name of the first corner point of the polygon.

B Name of the second corner point of the polygon.

etc.

A maximum of 26 corner points are allowed. The points are connected in the order in which they are entered in the PY card. The corner points have to be defined prior to the PY card by a DP card.

Example of PY card usage:

The commands below generate the polygon (in this case a triangle) shown, from an oblique angle, in figure 8-50 is created.

```
**
#height = 6
#width = 8
DP   A                          0          0         -#height/2
DP   B                         -#width/2   0          #height/2
DP   C                          #width/2   0          #height/2
PY   A    B    C
EG
EN
```

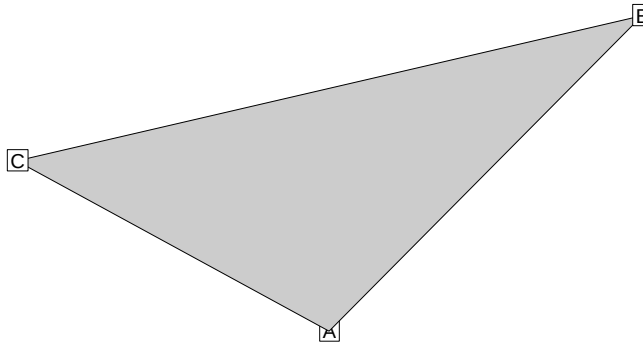


Figure 8-50: Example for the PY card

8.2.31 QU Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
QU		S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4	R_5	R_6		
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

Using this card a dielectric or magnetic cuboid, consisting of small cuboids, may be generated. Figure 8-51 shows a sketch.

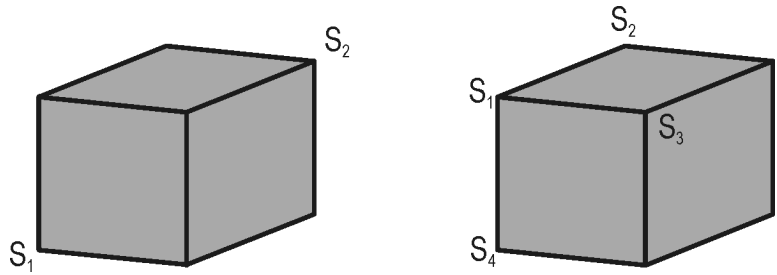


Figure 8-51: Sketch illustrating the use of the QU card

There are two possibilities:

- S_1, S_2 entered, S_3 and S_4 empty:
  A dielectric/magnetic cuboid, whose sides are parallel to the coordinate plane, is generated. Only S_1 and S_2, which are the coordinates of the opposite corners, are needed. This version is shown in figure 8-51 on the left.

- S_1, S_2, S_3 and S_4 entered:
  The cuboid may have an arbitrary orientation in space. S_1 is the reference point. From this point the other three points S_2, S_3 and S_4 are defined. They are three neighbouring corners of the cuboid. Care has to be taken to ensure that the cuboid is rectangular i.e. the lines $S_1 - S_2$, $S_1 - S_3$ and $S_1 - S_4$ must be perpendicular to each other. This version is shown in figure 8-51 on the right.

Parameters:

S_1	Name of any corner of the cuboid.
S_2	Second corner of the cuboid (see above).
S_3	Optional third corner of the cuboid (see above).
S_4	Optional fourth corner of the cuboid (see above).

only applicable to dielectric cuboids:

R_1 Relative dielectric constant ε_r of the dielectric cuboid.

R_2 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the cuboid.

R_3 Density ϱ in $\frac{\mathrm{kg}}{\mathrm{m}^3}$ of the cuboid. This value is only required for calculating the SAR (specific absorption rate), but must always be declared.

R_6 The electric loss tangent, $\tan(\delta)$

only applicable to magnetic cuboids:

R_4 Relative permeability μ_r of the magnetic cuboid.

R_5 Loss tangent $\tan\delta_\mu$ of the magnetic cuboid, where

$$\mu_r = \mu_r' - j\mu_r'' = \mu_r' \left(1 - j \tan\delta_\mu\right).$$

No dielectric bodies with surface equivalence principle and volume equivalence principle (using cuboids) can be used at the same time.

Example of QU card usage:

Using the following commands the dielectric cuboid, shown in figure 8-52 is generated.

```
**
IP                                                              1.1
DP   A                        1.0       0.0       0.0
DP   B                        7.0       3.0       1.5
QU   A    B                   5.0       0.0
EG
EN
```

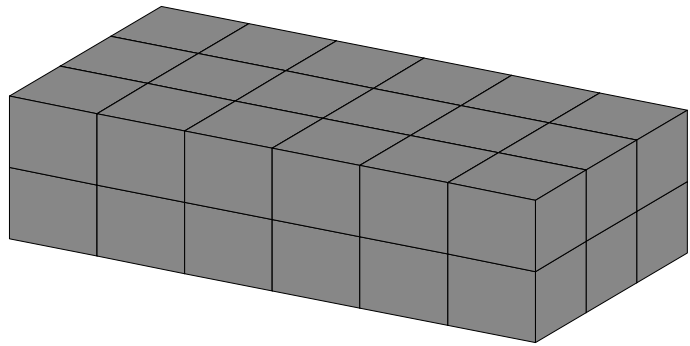
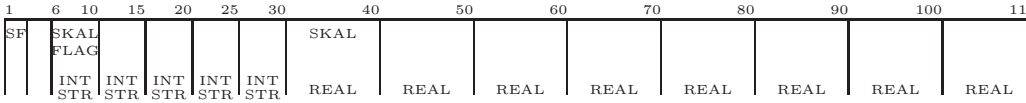


Figure 8-52: Example for the QU card

8.2.32 SF Card



With this card the scaling of the geometric data is possible.

Parameters:

SKALFLAG Determines what should be scaled.
- 0: All geometrical dimensions are scaled with the factor SKAL, but not the coordinates for near field calculations or excitations (see the discussion below).
- 1: All geometrical and coordinate dimensions are scaled.

SKAL The scaling factor (for example, if SKAL=0.001 all dimensions are entered in mm).

All geometrical dimensions and/or coordinates are multiplied by the factor SKAL. This is necessary when the input coordinates are not in metre, but, for example, in cm. (In this case the geometry has to be scaled by a factor of 10^{-2}.) Only one SF card is allowed in the input file. This is global and can be positioned anywhere. (However, since it is a geometry card it must be before the EG card.)

The mode SKALFLAG=0 is supported for compatibility with existing input files. For new input files it is strongly recommended to use the mode SKALFLAG=1. In the SKALFLAG=0 mode the following is scaled:

- Coordinates of the corner points of the triangular surface elements
- Coordinates of the corner points of the segments
- Radii of the segments
- Coordinates of the corner points of the cuboids
- Radii of the all the layers when the Green's function for a homogeneous or layered dielectric sphere is used (GF card, section 9.2.27)
- Thickness of the layers when the Green's function for a planar, multilayered substrate is used (GF card, section 9.2.27)
- Coordinates of the corner points of the polygonal plates
- Coordinates, radii and dimensions of UTD cylinders

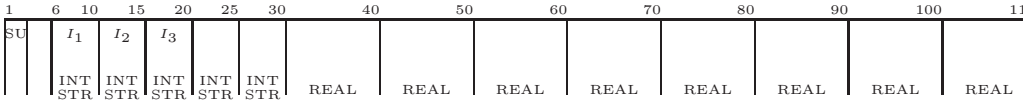
- Coordinates of the corner points of tetrahedral volume elements
- Thickness of dielectric surface elements (SK card, section 9.2.38)
- Radius and thickness of a wire coating (CO card, section 9.2.20)
- Coordinates of wedges and edges in the PO region
- Coordinates of the Fock region
- Transmission line length and end point coordinates (TL card, section 9.2.40)
- The dimensions of the aperture used in the AP card and the amplitudes of the A5 and A6 dipoles which depend on the incremental areas.
- The variable EPSENT, specified with the EG card (section 8.2.12), which controls whether two points are considered to lie at the same point is space

In the mode SKALFLAG=1 all geometrical dimensions and coordinates are scaled. This includes, in addition to the parameters listed above, the following:

- The coordinates of the source point specified in the excitation cards A1, A2, A3, A4, A5, A6, A7 (if the selection is not made by label)
- Coordinates of the origin of the radiation pattern specified with the AR card.
- Coordinates of the start and end points of the impressed currents for the AI and AV source cards, as well as the wire radius specified with these cards.
- Radii of the coaxial feed in the A3 card
- Radius of the approximated connecting segment in the A4 and L4 cards
- Positions where the near field is calculated with the FE card
- Offset in the near field calculation (OF card, section 9.2.34)

Note that if, for example, all data are specified in mm with SKAL=0.001, all input values are interpreted as mm. This also applies to the segmentation parameters (IP card) and possible translations (TG card).

8.2.33 SU Card



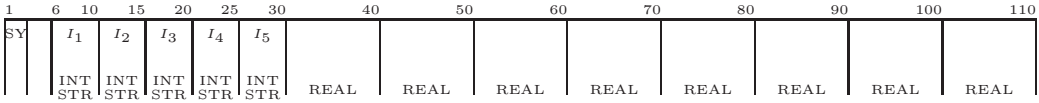
With this command the program runs in the superuser mode so that special parameters may be set.

Parameters:

- I_1 The password to enter super-user mode. *The super-user mode is only available to the code developers as it ignores the sensible defaults, warnings and errors given by FEKO.*
- I_2 This variable sets BLOCKNB, the block size for the solution of the matrix equation with ScaLAPACK (both in main memory, and using an auxiliary file). Typically a sensible value is allocated when the SU card is not used, so that this is only necessary when performance tuning is required for specific computational architecture. Note that I_2 can be set without setting I_1.
- I_3 Indicates the file system type for the parallel version of FEKO with the solution of the matrix equation when an auxiliary file is necessary (out-of-core ScaLAPACK):
  0: Default.
  1: Distributed (this means every process has its own file).
  2: Shared (a single file for all processes).
  3: Interlaced (only for testing; not stable).
  The location of the files can be set through the use of the environment variable `FEKO_TMPDIR` (see section 2.7). Note that I_3 can be set without setting I_1.

Certain parts of the program that are present for development and debugging, e.g. the use of symmetry in the SY card without saving computational time or the parts that still are under development and testing, can only be run in the superuser mode.

8.2.34 SY Card



Here symmetry can be used to generate the geometry and to reduce computation time.

Parameters:

I_1 1: Employ symmetry to save CPU time.
0: Mirroring the geometry, but not using the symmetry to save CPU time. This option is only available during program development and cannot be activated by normal users.

I_2 Mirroring around the plane $x = 0$ (or yz plane).
1: The plane $x = 0$ represents geometric symmetry.
2: The plane $x = 0$ represents electric symmetry.
3: The plane $x = 0$ represents magnetic symmetry.

I_3 Mirroring around the plane $y = 0$ (xz plane).
1: The plane $y = 0$ represents geometric symmetry.
2: The plane $y = 0$ represents electric symmetry.
3: The plane $y = 0$ represents magnetic symmetry.

I_4 Mirroring around the plane $z = 0$ (xy plane).
1: The plane $z = 0$ represents geometric symmetry.
2: The plane $z = 0$ represents electric symmetry.
3: The plane $z = 0$ represents magnetic symmetry.

I_5 After being mirrored, the mirrored parts are given a label, it is the previous label with the value of I_5 added to it. An exception is label 0; the corresponding new parts will also have label 0.

All the conducting and/or dielectric triangles, segments, cuboids, tetrahedral volume elements, wedges, edges, Fock regions and polygonal surfaces that have been declared before the SY card, are mirrored. Furthermore, the second and third corners of the triangles are swapped, such that the direction of the normal vector is retained. Likewise the corners of image polygons are rearranged to retain the normal direction. (The first corner point of the original polygon becomes the last corner of the mirror image.)

Sources are not mirrored. If, for example, a Hertzian dipole is placed on one side of the symmetry plane, the user must also place the correct image on the opposite side of the symmetry plane.

Multiple SY cards can be used and it is possible to mirror around more than one plane at once (e.g. I_1=I_2=I_3=1). A detailed description of the different types of symmetry (geometric, electric and magnetic symmetry) was given in section 2.4.

8.2.35 TG Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
TG		I_1	I_2	I_3	I_4	I_5	R_1	R_2	R_3	R_4	R_5	R_6	R_7	
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this command the already entered geometric elements (triangles, segments, etc.) can be translated, rotated and/or scaled. It is also possible to duplicate structures.

Parameters:

I_1 For I_1=0 the geometry that has been created up to where the TG card is entered, is translated, rotated and/or scaled. The number of triangles and segments remains the same.

For I_1 not equal to zero, the values represents the number of new structures to be generated, e.g. for I_1=3 and R_3=90° an antenna can be rotated 3 times around the z axis so that there will be a total of 4 antennas. The total number of segments and triangles will now be the number that was present before the TG card multiplied by (I_1+1).

I_2,I_3 Labels can be used to select a certain part of the structure (see also selection with the parameter I_5) that is to be rotated, translated or scaled. The TG card will be applied only to those parts of the structure whose label lies within the range $I_2 \leq$ label $\leq I_3$ (see LA and CB cards).

I_4 Each newly generated structure will be assigned a label that is incremented by I_4 from that of the original structure. An exception is the label 0 which is retained.

I_5 Selection of the structure that is affected by the rotation, translation and/or scaling. This applies to:

- 0: All structures already entered (metallic/dielectric triangles, metallic segments, dielectric/magnetic cuboids, wedges and edges in PO regions, Fock regions, polygonal plates, UTD-cylinders and tetrahedral elements) are processed. The label-criterion of specified by the parameters I_2 and I_3 remains in effect.
- 1: Only metallic triangles, Fock regions, and wedges and edges in PO regions that satisfy the label-criterion of I_2 and I_3, are processed.
- 2: Only metallic segments that satisfy the label-criterion of I_2 and I_3, are processed.
- 4: All dielectric triangles are processed. Labels are ignored and selection with I_2 and I_3 is not possible. This is only supported for compatibility reasons and the use I_5=32 is preferred.

8: All dielectric and magnetic cuboids are processed. Labels are ignored and selection with I_2 and I_3 is not possible. This is only supported for compatibility reasons and the use I_5=64 is preferred.

16: All polygonal plates are processed. Labels are ignored and selection with I_2 and I_3 is not possible. This is only supported for compatibility reasons and the use I_5=128 is preferred.

32: Only dielectric triangles that satisfy the label-criterion of I_2 and I_3, are processed.

64: Only dielectric and magnetic cuboids that satisfy the label-criterion of I_2 and I_3, are processed.

128: Only polygonal plates that satisfy the label-criterion of I_2 and I_3, are processed.

256: Only UTD-cylinders that satisfy the label-criterion of I_2 and I_3, are processed.

512: Tetrahedral volume elements that satisfy the label-criterion of I_2 and I_3, are processed.

I_5 can assume the tabulated values directly. It is also possible to add the options in a binary fashion. For example, setting I_5=1+2=3 all metallic triangles and segments (with the correct label) are processed.

R_1 Angle of rotation α_x around the x axis in degrees.

R_2 Angle of rotation α_y around the y axis in degrees.

R_3 Angle of rotation α_z around the z axis in degrees.

R_4 Translation Δ_x in the x direction in m (scaled by SF card).

R_5 Translation Δ_y in the y direction in m (scaled by SF card).

R_6 Translation Δ_z in the z direction in m (scaled by SF card).

R_7 The scaling factor γ, with which the structures must be scaled (if the parameter R_7 is not specified, it defaults to $\gamma = 1$). For wire segments the wire radius is scaled as well as the coordinates of the start and end points.

When an SY card (symmetry) is used before the TG card, the TG card resets the symmetry if the new structures invalidates the symmetry. Cases where the symmetry is not reset is when, for example, the plane $z = 0$ is a symmetry plane and the TG card specifies rotation about the z axis for a symmetrical selection of elements. In this case the symmetry is retained.

With a TG card the simultaneous rotation around multiple axes as well as translation in multiple directions is possible. A point (x, y, z), for example the corner point of a

triangle, is transformed to a new point

$$\begin{bmatrix} x_T \\ y_T \\ z_T \end{bmatrix} = \gamma\, M \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \gamma \begin{bmatrix} \Delta_x \\ \Delta_y \\ \Delta_z \end{bmatrix}$$

with the rotation matrix

$$M = \begin{bmatrix} \cos\alpha_y \cos\alpha_z & -\cos\alpha_y \sin\alpha_z & \sin\alpha_y \\ \cos\alpha_x \sin\alpha_z + \sin\alpha_x \sin\alpha_y \cos\alpha_z & \cos\alpha_x \cos\alpha_z - \sin\alpha_x \sin\alpha_y \sin\alpha_z & -\sin\alpha_x \cos\alpha_y \\ \sin\alpha_x \sin\alpha_z - \cos\alpha_x \sin\alpha_y \cos\alpha_z & \sin\alpha_x \cos\alpha_z + \cos\alpha_x \sin\alpha_y \sin\alpha_z & \cos\alpha_x \cos\alpha_y \end{bmatrix}$$

Multiplication by the rotation matrix M effectively rotates a point first by an angle α_z around the z axis, then by an angle α_y around the y axis and finally by an angle α_x around x axis. If more than one copy is made, successive points are generated from the previous point using the same relation.

The file `example_18.pre` (see the *Examples Guide*) gives an example of an application of the TG card.

8.2.36 TO Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
TO	S_1	S_2	S_3	S_4		R_1	R_2	R_3	R_4	R_5	R_6			
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

Using this card a toroidal segment can be generated. Figure 8-53 shows a sketch.

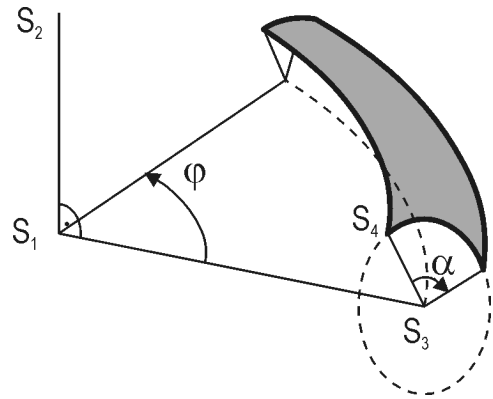


Figure 8-53: Sketch illustrating the use of the TO card

Parameters:

- S_1 Name of the centre of the toroid.
- S_2 The name of a point that is perpendicular and is situated an arbitrary distance above S_1 which is in the plane of the toroid.
- S_3 The name of the point that represents the axis (see figure 8-53).
- S_4 A point situated on the surface on the toroidal segment. It must be in the plane formed by the three points S_1, S_2 and S_3.
- R_1 The angle φ in degrees of rotation around the axis S_1–S_2.
- R_2 The angle α in degrees (see figure 8-53).
- R_3 The maximum edge length along the curved edge in the φ direction in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.
- R_4 The maximum edge length along the curved edge in the α direction in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.
- R_5 Indicates the direction of the normal vector:
  0: Normal vector points to exterior.
  1: Normal vector points to interior.

R_6 If this parameter is empty or is set to 1, a toroid with a circular cross section is created. The parameter R_6 may be used to generate an elliptical cross section in the α plane. $R_6 = \frac{b}{a}$ gives the ratio of the ellipse's two half axes, where a is the distance S_3–S_4. It is recommended to generate toroids where the elliptical cross section have extremely small or extremely large axial ratios with a CAD system (such as FEMAP) as the distortion formulation used in PREFEKO may fail in these cases.

A complete toroid is obtained by using the parameters $\varphi = 360°$ and $\alpha = 360°$. The normal vector generated by the triangles points to the outside.

First example of TO card usage:

Using the following commands the toroidal segment, which is shown in figure 8-54, is generated.

```
**
IP                                          0.7
DP   A                          0.0         0.0         0.0
DP   B                          0.0         0.0         1.0
DP   C                          1.0         0.0         0.0
DP   D                          1.0         0.0         0.2
TO   A    B    C    D          135.0       360.0       0.2         0.15
EG
EN
```

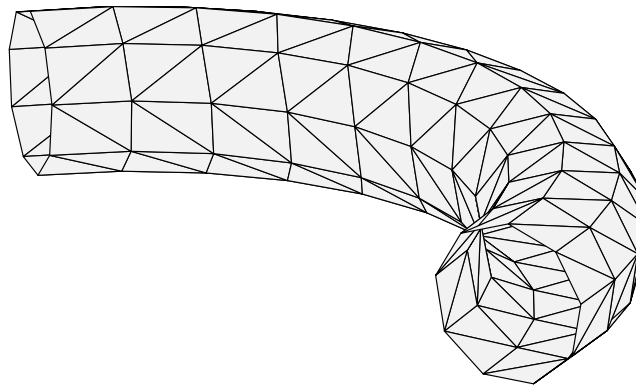


Figure 8-54: Example for the TO card

Second example of TO card usage:

The commands below generates the toroidal segment with an elliptical cross section as shown in figure 8-55.

```
** example of an elliptical torus
**
IP                                        0.7
DP   A                        0.0         0.0         0.0
DP   B                        0.0         0.0         1.0
DP   C                        1.0         0.0         0.0
DP   D                        1.0         0.0         0.2
TO   A    B    C    D         90.0        360.0       0.2         0.15        0           2.0
EG
EN
```

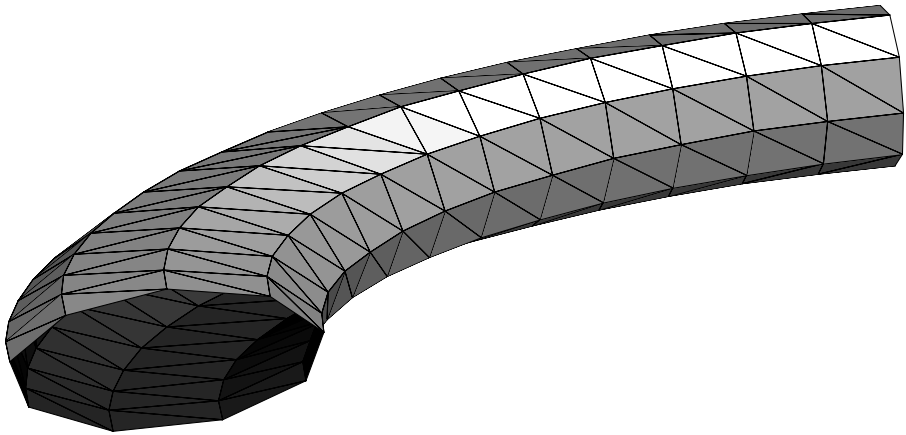
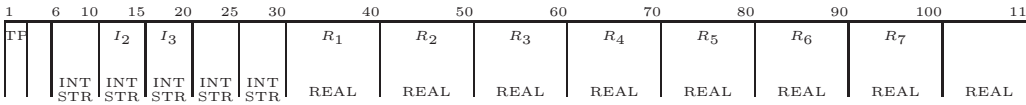


Figure 8-55: Example for the TO card with an elliptical cross section (R_6 is specified)

8.2.37 TP Card



With this card points (previously defined with the DP card) can be translated, rotated and/or scaled.

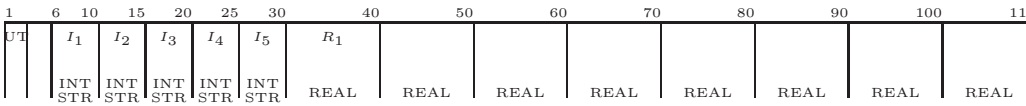
Parameters:

- I_2 All points with labels in the range $I_2 \leq$ label $\leq I_3$ are translated, rotated or scaled. The parameter I_2 specifies the start label.
- I_3 The end of the label range.
- I_4 Each transformed point will be assigned a label that is the label of the original point incremented by I_4. The exception are points with label 0 — their label is not incremented, it remains 0.
- I_5 For I_5=0 the scaling factor R_7 is applied to all three coordinates of each point. For I_5 <> 0, scaling is done by interpreting I_5 as a 3-bit number with possible values from 1 to 7. If the first bit (value 1) is set, then the scaling factor R_7 does not apply to the x coordinate of the point. If the second bit (value 2) is set, the y coordinate is not scaled, and if the third bit (value 4) is set, then the z coordinate is not scaled. If $I_5 = 7$ no scaling is done (and then one must set $R_7 = 1$).
- R_1 Angle of rotation α_x around the x axis in degrees.
- R_2 Angle of rotation α_y around the y axis in degrees.
- R_3 Angle of rotation α_z around the z axis in degrees.
- R_4 Translation Δ_x in the x direction. (All three translation distances are affected by the scaling factor set with the SF card.)
- R_5 Translation Δ_y in the y direction.
- R_6 Translation Δ_z in the z direction.
- R_7 The scaling factor γ, with which the point is scaled after rotation and translation (if the parameter R_7 is not specified, it defaults to $\gamma = 1$).

If a point is rotated around more than one axis with a single card, it is rotated first by an angle α_z around z axis, then by α_y around the y axis and finally by α_x around the x axis. A more detailed description of the transformation can be found in the description of the TG card (section 8.2.35).

In an exception to the rule that all geometry cards must appear before the EG card, this card (as well as the DP card) can be used after the EG to define points for use in the AP card.

8.2.38 UT Card


1 6 10 15 20 25 30 40 50 60 70 80 90 100 110
UT
I_1 I_2 I_3 I_4 I_5 R_1
INT INT INT INT INT REAL REAL REAL REAL REAL REAL REAL REAL
STR STR STR STR STR

With this command the parameters for the geometric theory of diffraction (UTD) for the polygonal surfaces are defined.

Parameters:

I_1 The parameter I_1 (UTDFLAG) determines whether the classical GTD by Keller (I_1=0) or whether the uniform diffraction theory UTD by Kouyoumjian (I_1=1) should be used. It is advisable to always use the UTD (I_1=1), because singularities appear on the shadow boundaries in Keller's approach.

I_2 The parameter I_2 (UTDMAXRX) gives the maximal number of ray-interactions (i.e. reflection and diffraction combined). With, for example, I_2=3 a ray can have 3 reflections, or 2 reflections and a diffraction. For I_2=0 only direct rays are taken into account.

I_3 For I_3=1 a debug file (extension `*.dbg`) is generated. This file contains large amounts of information and should only be used when debugging. With I_3=0 no debug file is generated.

I_4 For I_4=1 the ray information is exported to the `*.bof` and to a special `*.ray` file, so that the ray paths can be displayed in WinFEKO. The ray information can become very large, and thus it should only be exported if specific ray paths are to be examined. For I_4=0 no ray information is exported.

I_5 Determines which ray contributions to take into account:

- 1: Geometric optics (GO), i.e. direct and reflected rays and shadow regions are taken into account.
- 2: Diffraction on edges and wedges as well as reflections and a diffraction are taken into account.
- 4: Corner diffraction.
- 8: Double diffraction on edges and wedges and combinations of reflections are taken into account.
- 16: Creeping waves on curved surfaces.
- 32: Tip diffraction at the tip of a cone.

A value of I_5=7=1+2+4 means that e.g., the UTD is applied using reflections and a simple edge and wedge diffraction as well as a corner point diffraction are used. In normal cases I_5 together with I_2 should be a compromise between accuracy and computational time.

R_1 The variable R_1 (UTDNOCOUPL) specifies whether the coupling from the UTD region to the MoM region should be considered:

0: The usual option, that is the coupling of the MoM region with the UTD region is considered.

1: The coupling of the MoM region with the UTD region is neglected, that is, there is no coupling back to the MoM region from the UTD region (shadowing, reflection). This option is especially relevant when the MoM region and the UTD region is not situated close together.

When no UT card is used, the following default values apply:

$I_1 = 1$
$I_2 = 3$
$I_3 = 0$
$I_4 = 0$
$I_5 = 7$
$R_1 = 0$

The following restrictions apply for the hybrid MoM/UTD:

- no dielectric bodies or dielectric ground
- only perfectly conducting flat polygonal plates or a single cylinder allowed in the UTD region
- no UTD and PO at the same time

8.2.39 UZ Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
UZ		S_1	S_2	S_3			R_1	R_2	R_3	R_4	R_5			
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card a cylinder is created for the UTD region, figure 8-56 shows a sketch.

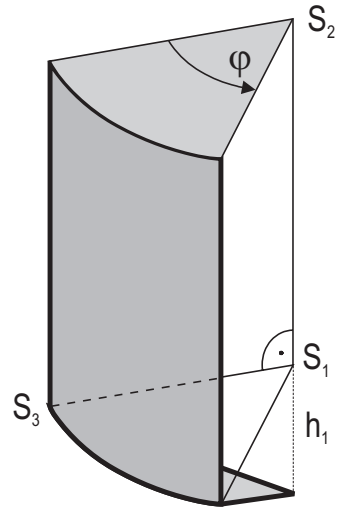


Figure 8-56: Sketch for UZ card with a cone shaped end cap, height h_1 from point S_1 and flat surface end cap at point S_2.

Parameters:

- S_1 Name of the starting point of the cylinder axis
- S_2 Name of the end point of the cylinder axis
- S_3 Name of a point on the rim at the side of S_1. For a cylinder segment, this is the corner point.
- R_1 The angle φ in degrees of the cylinder segment
- R_2 Flag for the type of end cap on the S_1 side. It has the following meaning:
  - -1: No end cap, thus the cylinder is semi-infinite on this side.
  - 0: Flat end cap in the form of a disk sector.
  - 1: Spherical end cap.
  - 2: Elliptical end cap with height $h_1 = R_3$.
  - 3: Cone shaped end cap with height $h_2 = R_3$.

R_3 Height of the end cap on the S_1 side (only applicable when $R_2 = 2$ or $R_2 = 3$) in m (is scaled by the SF card).

R_4 Flag for the type of end cap on the S_2 side. It has the same meaning as R_2.

R_5 Height of the end cap on the S_2 (only applicable when $R_4 = 2$ or $R_4 = 3$) in m (is scaled by the SF card).

Example of UZ card usage:

The following commands generate the cylinder in figure 8-57

```
** Example of a cylinder in UTD region
DP   A                          0         0        -2
DP   B                          0         0         2
DP   C                          1         0        -2
UZ   A    B    C                360       0                    0
EG
EN
```

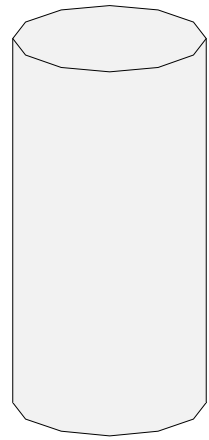
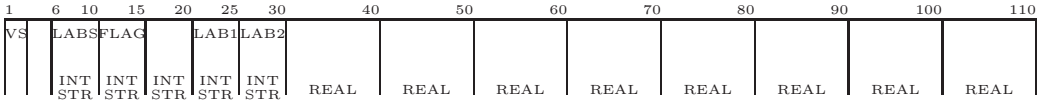


Figure 8-57: Example for UZ card

8.2.40 VS Card



This card specifies visibility information to reduce the time required to calculate the visibility information required when using physical optics with multiple reflections.

To accurately compute multiple reflections the code needs to determine which basis functions are visible to each other. For large problems this may be very time consuming. The time required to determine the visibility may be greatly reduced if the user can inform the code that certain triangles are hidden from each other and others are visible to each other.

Parameters:

LABS	The label of the source triangles.
FLAG	1: All triangles with label LAB1 are visible from all triangles with label LABS. 2: All triangles with labels in the range LAB1 to LAB2 (inclusive) are visible from all triangles with label LABS. 3: All triangles with label LAB1 are hidden from all triangles with label LABS. 4: All triangles with labels in the range LAB1 to LAB2 (inclusive) are hidden from all triangles with label LABS.
LAB1	The label of the visible or hidden triangles.
LAB2	The end of the label range if FLAG is 2 or 4.

Note that visibility is reciprocal, i.e. if all triangles with label LAB1 are visible from all triangles with label LABS, all triangles with label LABS are visible from all triangles with label LAB1 as well.

The labels must be specified in ascending order, i.e. LABS $\leq$ LAB1 $<$ LAB2. LABS may never decrease in consecutive VS cards. In addition, for different VS cards with the same value of LABS, LAB1 of any card must be larger than LAB1 (in case of a single layer) or LAB2 (in case of a range of layers) of the previous card. (See the example below.)

Basis functions cannot illuminate each other if all the triangles they are attached to lie in the same plane.

The VS card should only be used if the user can specify the visibility beyond any doubt. If no information is specified for a specific combination of labels/triangles, full ray tracing will be executed.

Example of VS card usage:

Consider the structure shown in figure 8-58 consisting of four flat plates and a cylindrical section. The two plates lying at 45 degrees to the coordinate system (labelled 1 and 3), are half as wide as the plates with labels 0 and 2. Thus some triangles with label 2 are visible to some triangles with label 0, but not all.

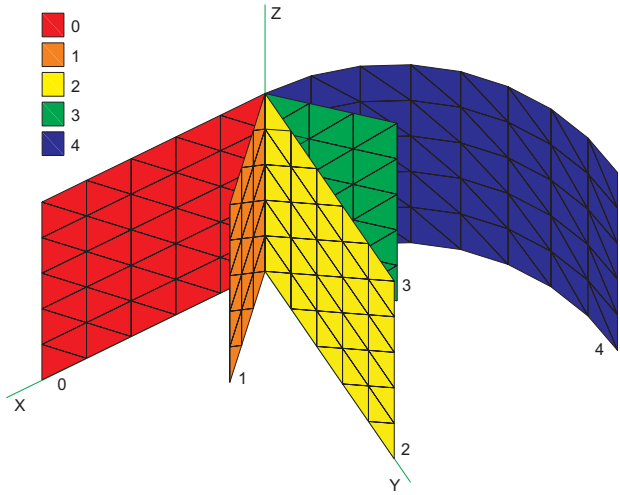


Figure 8-58: Structure used to demonstrate the use of VS cards

We have to specify which triangles are visible\hidden from *all* triangles with label 0 first, then those visible from label 1 and so on. The VS cards for this example would be as follows:

```
VS    0    3          0
VS    0    1          1
VS    0    4          3    4
VS    1    3          1
VS    1    1          2
VS    1    4          3    4
VS    2    3          2
VS    2    1          3
VS    3    3          3
VS    3    1          4
```

Since all the triangles with label 0 lie in the same plane, they cannot illuminate each other. Thus the first card states that label 0 is hidden from label 0.

All triangles with label 1 are visible from all triangles with label 0. This is specified by the second VS card. Since some triangles with label 2 are visible from some triangles with label 0 while others are hidden, we cannot specify any information for this combination of layers. However, the plate with label 2 shadows all triangles with labels 3 and 4 and we may specify that these are hidden. This is done with the third VS card. Note that this card specifies a range of hidden labels.

Next we must specify which triangles are visible (or hidden) from *all* triangles with label 1. As for label 0, triangles with label 1 are not visible to each other — specified by the fourth VS card. All triangles with labels 0 and 2 are visible from all triangles with label 1. Since we have already specified the visibility between labels 0 and 1, we do not specify it again. The fifth VS card then specifies that label 2 is completely visible from label 1. As for label 0, both labels 3 and 4 are hidden completely which completes the first six VS cards,

Next we look at label 2. As before we need not consider labels lower than 2. Also the label is hidden from itself as indicated by VS card number seven. Next we state that label 3 is visible, but we cannot specify anything about label 4 as only some of these triangles will be visible.

Similarly VS cards 9 and 10 states that label 3 is not visible to itself and fully visible to label 4.

Finally we must consider the case for triangles with label 4. All visibility with layers 0 to 3 has been specified and may not be specified again. Unlike the previous flat plates, layer 4 is curved and some triangles may indeed illuminate other triangles with the same layer. However, not all other triangles will be illuminated (this is only possible for a doubly concave surface), such that we cannot specify any information for label 4.

8.2.41 WG Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
WG	S_1	S_2	S_3	S_4			R_1	R_2						
	INT STR	INT STR	INT STR	INT STR	INT STR		REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card a wire grid in the shape of a parallelogram can be generated. In figure 8-59 a sketch is shown.

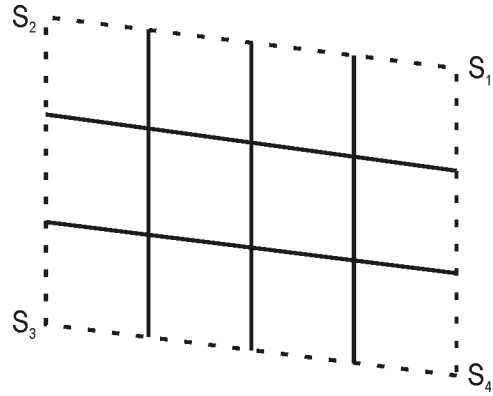


Figure 8-59: Sketch illustrating the use of the WG card

Parameters:

- S_1 Name of the first corner point of the parallelogram.
- S_2 Name of the second corner point of the parallelogram.
- S_3 Name of the third corner point of the parallelogram.
- S_4 Name of the fourth corner point of the parallelogram.
- R_1 0: Only the wires inside the parallelogram are generated.
  1: All the wires are generated. This option is important when two adjacent parallelograms are generated, because the sides must then not be generated twice.
- R_2 The maximum segment length is given by the IP card. The parameter R_2 is an integer number and represents the length of the gaps in the wire grid. For $R_2 = 1$ the gaps are one segment in length, for $R_2 = 2$ two segments, etc.

Example of WG card usage:

Using the following commands the wire grid seen in figure 8-60 is generated.

```
**
IP                                  0.02                0.2
DP   A                              0.0       0.0       0.0
DP   B                              0.0       0.0       1.2
DP   C                              1.2       0.0       1.2
DP   D                              1.2       0.0       0.0
WG   A    B    C    D               1         1
EG
EN
```

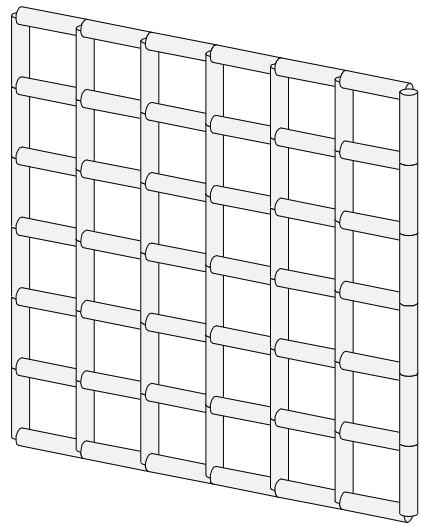


Figure 8-60: First example for the WG card

8.2.42 ZY Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
ZY		S_1	S_2	S_3	S_4		R_1	R_2	R_3					
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card a cylindrical segment can be generated. A sketch is shown in figure 8-61.

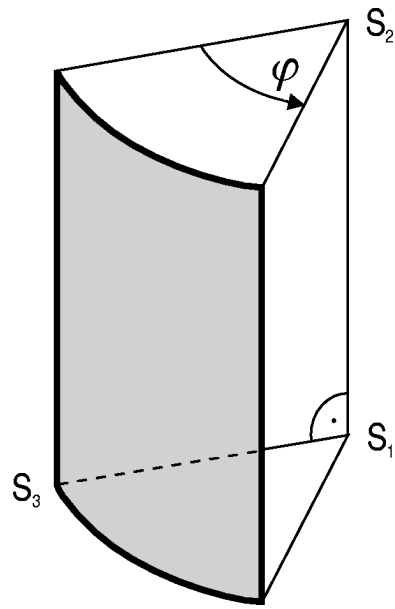


Figure 8-61: Sketch illustrating the use of the ZY card

Parameters:

- S_1 Name of the begin point of the axis.
- S_2 Name of the end point of the axis.
- S_3 Name of a point on the corner of the cylindrical segment.
- S_4 0: The normal vector of the triangles points outward (away from the cylinder).
  1: The normal vector points to the inside.
- R_1 The angle φ in degrees, which is subtended by the cylindrical arc.
- R_2 Maximum edge length of the triangles along the curved side in m (is scaled by the SF card). If this parameter is left empty, the value specified with the IP card is used.

R_3 If this parameter is empty or is set to 1, a circular cylinder is created. Within certain limits, the parameter R_3 may be used to generate an elliptical cylinder. $R_3 = \frac{b}{a}$ gives the ratio of the two half axes, where a is the distance S_1–S_3. It is recommended to generate elliptical cylinders with extremely small or extremely large axial ratios with a CAD system (such as FEMAP) as the distortion formulation used in PREFEKO may fail in these cases.

The segmented area (shaded in figure 8-61) is obtained by rotating the point S_3 around the axis S_1–S_2 through the angle φ. For $\varphi = 360°$ a full cylinder is obtained.

The fineness of the mesh on the sides parallel to the axis is determined by the value specified with the IP card. The parameter R_2 determines the maximum edge length along the curved side such that a better representation of the curve may be obtained if required. If the geometry representation is accurate enough, R_2 may be left empty and the value specified by the IP card will be used also along the curved edges.

The direction of the normal vector is obtained through the parameter S_4.

First example of ZY card usage:

Using the following commands the cylindrical section, shown in figure 8-62, is generated.

```
**
IP                                          0.4
DP   A                          0.0         0.0         0.0
DP   B                          0.0         0.0         2.0
DP   C                          1.0         0.0         0.0
ZY   A    B    C                180.0       0.35
EG
EN
```

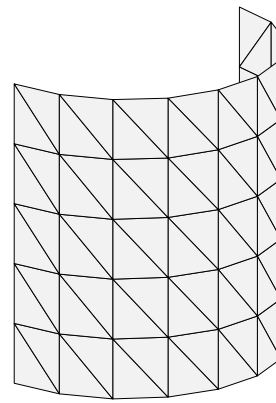


Figure 8-62: Example for the ZY card

Second example of ZY card usage:

An elliptical cylinder, as shown in figure 8-63, is generated with the following commands.

```
** Example for an elliptical cylinder

** General parameters
#a   = 1.5    ** first half axis of the elliptical cylinder
#b   = 2.5    ** second half axis of the elliptical cylinder
#h   = 4      ** height of the cylinder

** Segmentation
#kanl = 0.4
IP                                      #kanl

** Define points
DP   A                          0         0         0
DP   B                          0         0         #h/2
DP   C                          #a        0         0

** Define the geometry
ZY   A    B    C                90        #kanl     #b/#a

** End
EG
EN
```

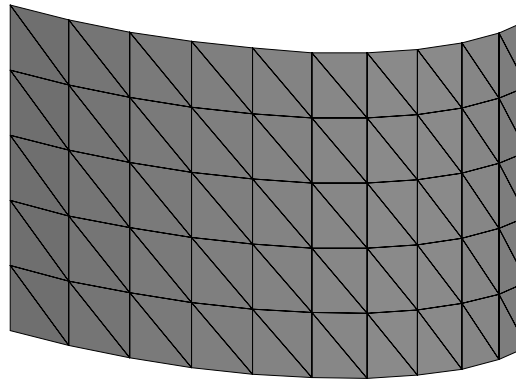


Figure 8-63: Example of an elliptical cylinder

9 Description of the control cards

9.1 Overview of control cards and remarks on execution sequence

The following table summarises all the PREFEKO and FEKO control cards. These cards should not be used in the geometry section of the `*.pre` input file, i.e. they should only be used below the EG card. The control cards are used to specify, for example, the frequency and the type of excitation. They also determine the required calculations (such as the locations for near field calculations and the directions for far fields calculations, etc.).

Card	Description
**	characters used to indicate a comment line
Ax	type of excitation (e.g. an incident plane wave or a voltage source)
BO	through the use of the reflection factor coefficient a ground plane can be inserted
CG	the algorithm used to solve the matrix equation is selected
CM	Field calculation for CableMod (coupling into transmission lines)
CO	inserts a dielectric and/or magnetic surface on the elements
DA	creates additional files for the results
DI	enters the properties of the dielectric when using the surface current method
EN	indicates the end of the input file
FE	calculates the near fields
FF	calculates the far fields
FR	sets the frequencies at which the calculations are to be carried out
GF	sets the Greens functions
L4	add a load between a metallic triangle and the ground plane for the planar multilayer Greens function
LD	defines a distributed series circuit load, consisting of resistance, inductance and capacitance
LE	defines a load on the edge between surface triangles
LP	defines a parallel circuit load, consisting of resistance, inductance and capacitance
LS	defines a series circuit load, consisting of resistance, inductance and capacitance
LZ	defines a complex load
OF	offset i.e. displacement of the origin when calculating the near fields
OS	saves the surface currents in a file
PS	set general control parameters
PW	defines the radiating power of a transmitting antenna
SK	takes a finite conductivity into account through the skin effect of ohmic losses; also for thin dielectric layers
SP	calculates the S-parameters for the active sources
TL	specifies a non-radiating transmission line

As mentioned above the control cards form the second part of the input file (see also section 2.1). Control cards are processed line by line and only affect other cards and calculations specified below them in the input file. (Information specified in a control card is not available before that line is processed.) Any number of control cards can be used, but they should adhere to a basic sequence. Thus, for example, the frequency (FR card) and the type of excitation (Ax card) must be defined before the near fields can be calculated with an FE card.

In addition, a sensible order for the control cards can result in a considerable reduction in the computation time. For example, for an SK card the whole matrix has to be recalculated, while an Ax card only redefines the right hand side of the matrix. A summary of the dependencies is given in the table below:

Action	For the cards
recalculates matrix elements	BO, CO, DI, FR, GF, LD, LE, LP, LS, LZ, SK, TL
recalculates the right hand side	Ax, BO, DI, FR, GF
resolves the matrix equation	CG

There are also other dependencies. If the matrix elements are recomputed then the matrix equation has to be solved again. The actual calculation is started by the FE, FF, OS and SP cards. All other cards are read and the data stored.

When solving for a number of frequencies (parameter NFREQ in the FR card) all the control cards following the FR card (until the next FR card or EN card) are read into a buffer. For each frequency all these cards are read and processed. The computation time can be reduced significantly by using the cards in the correct order. If, for example, a structure needs to be investigated at two frequencies and with two different excitations then the control cards can be organised in either of the following ways:

```
...
FR    FR card for the two frequencies
Ax    first excitation
...
Ax    second excitation
...
EN    end of the input file
```

or

```
...
FR    FR card for the two frequencies
Ax    first excitation
...
FR    FR card for the two frequencies
Ax    second excitation
...
EN    end of the input file
```

In the first format, the two excitations are processed one after the other for each frequency. (The cards are executed in the order FR Ax ... Ax ... FR Ax ... Ax ... EN, where the second FR indicates execution of the frequency loop for the second specified frequency.) In the second format the first excitation is treated for both frequencies before treating the second excitation for both frequencies. (Here the cards are executed as FR Ax ... FR Ax ... FR Ax ... FR Ax ... EN.) For the computational requirements, one finds that in the first case the matrix elements have to be calculated twice (it have to be completely recalculated each time the frequency is changed) and in the second case four times.

The computation time is not only influenced by the control cards, but also by what has to be solved for. In the following example, the structure has to be solved at a number of frequencies and for the ideal (conducting) and non-ideal (conduction with losses) cases:

```
...
FR    FR card for multiple frequencies
FE    calculation of the near fields
SK    include skin effect
FE    calculation of the near fields
EN    end of the input file
```

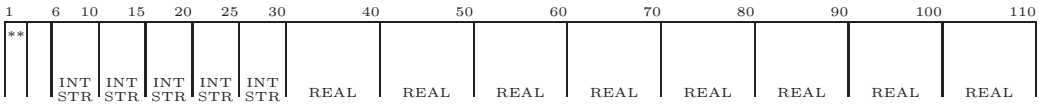
The three control cards FE, SK and FE are written to the buffer and are worked through in the loop for the different frequencies. At the first frequency the FE card initiates the field calculation. Because a SK card has not yet been read, ideal conductivity is assumed. Then the SK card is read and losses are taken into account when the second FE card is run. Thus the first frequency pass is finished. At the next frequency pass the cards FE, SK, and FE are read again, but the losses from the SK card are still active from the first pass. The SK card is thus useless and the two FE cards calculate the same things twice. The correct input order is:

```
...
FR    FR card for multiple frequencies
SK    skin effect switched off
FE    calculation of the near fields
SK    skin effect switched on
FE    calculation of the near fields
EN    end of the input file
```

Then the four cards SK, FE, SK and FE are calculated for all the frequencies.

9.2 Detailed description of the control cards

9.2.1 ** Card



The comment lines discussed in section 8.2.1 can also be used after the EG card.

9.2.2 Ax Cards

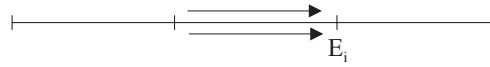
This card defines the type of excitation as well as other parameters regarding the excitation. The following possibilities are available:

Card	Type of Excitation
A0	A linear polarised plane wave incident on the structure.
A1	Excitation by means of a voltage gap on a segment (i.e. impressed electric field strength along a segment).
A2	Excitation by means of a voltage gap at a node i.e. between two segments
A3	Excitation by means of a magnetic ring current (TEM-Frill) on a segment. Thus a coaxial feed can be modelled.
A4	Special vertical pin excitation, e.g. for a patch antenna on a planar substrate with a ground plane (coaxial probe excitation mode)
A5	A Hertzian dipole is used as excitation. The position and orientation in the space are arbitrary.
A6	A Magnetic dipole is used as excitation. The position and orientation in the space are arbitrary.
A7	Excitation by means of a voltage gap on an edge between two triangles
AC	This card reads the geometry and current distribution (possibly for more than one frequency) from an *.rsd file created by the transmission line simulation program CableMod[9] or with the OS card in FEKO. The excitation is due to the electromagnetic fields radiated by these line currents.
AE	The AE card is an excitation between triangle edges similar to the A7 card, however the AE card permits the simultaneous excitation of several edges.
AI	Excitation by an impressed line current.
AP	Excitation with an aperture (array of electrical and magnetic Hertzian dipoles)
AR	Excitation by an antenna with a given radiation pattern.
AV	Excitation by an impressed line current similar to the AI card, but the endpoint of the current is electrically connected to a conducting surface.

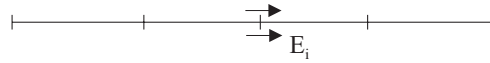
The different ways to realise a voltage source are summarised in figures 9-1 and 9-2. The impressed electric field strength is indicated by $\vec{E}_i$.

[9] To use the CableMod interface this module must be activated, if required please contact EMSS.

A1 card: Voltage source on a segment



A2 card: Voltage source on a node between segments



A3 card: TEM-frill on a segment

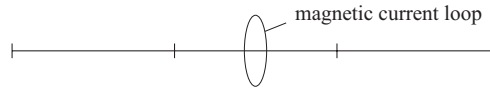
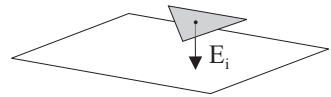
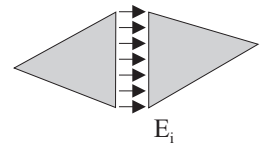


Figure 9-1: Possible ways to realise a voltage source on a wire segment

A4 card: Vertical pin approximation (dielectric substrate)



A7 card: Voltage gap on an edge



AE card: Voltage gap along edges

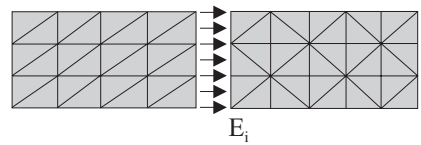


Figure 9-2: Possible ways to realise a voltage source in connection with triangles

More than one excitation is also allowed. Thus one may, for example, generate an elliptically polarised plane wave by super-imposing two out of phase linearly polarised plane waves with different amplitudes. It is also possible to feed an antenna with two different voltage sources. For this purpose the parameter ANFL is available in each Ax card. This parameter indicates whether the current excitation is additional (ANFL=1) or not (ANFL=0). When ANFL=0 only the current excitation will be used and the excitations prior to the current one will be erased.

For the excitations A1, A2, A3, A4 and A7 it is possible to select the feed element through the label. Alternatively the label parameter ULA can be set to -1, then the position of the feed is specified in Cartesian coordinates. This should, simplify the modelling in most cases — especially with the A7 card. FEKO then search for a segment or an element at this position (with the A7 card the specified point must be in the middle of the edge). This comparison of the position uses the tolerance parameter EPSENT (EG card, section 8.2.12).

The excitations are described in detail in the following sections.

9.2.3 A0 Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
A0		ANFL		NTHEI	NPHII		EIR1	EIR2	EIR3	EIR4	EIR5	DTHEI	DPHII	EIR8
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card realises excitation by a linearly polarised incident plane wave.

Parameters:

ANFL
: 0: New excitation, replaces all previous excitations.
1: Additional excitation, add to previous excitations.

NTHEI
: If more than one direction of incidence is to be examined, then this parameter indicates the number of incident directions points in the ϑ direction. If this field is left empty or a 0 is entered, then NTHEI=1 is set.

NPHII
: If more than one direction of incidence is to be examined, then this parameter indicates the number of incident directions in the φ direction. If this field is left empty or a 0 is entered, then NPHII=1 is set.

EIR1
: Value of the field strength $\vec{E}_0$ of the incident field in $\frac{\mathrm{V}}{\mathrm{m}}$.

EIR2
: Phase of the field strength $\vec{E}_0$ of the incident field in degrees.

EIR3
: Angle of incidence ϑ of the plane wave in degrees, see figure 9-3.

EIR4
: Angle in incidence φ of the plane wave in degrees, see figure 9-3.

EIR5
: Polarisation angle η in degrees, see figure 9-3.

DTHEI
: If more than one direction of incidence is to be examined, then this value is the increment in the direction of incidence in the ϑ direction.

DPHII
: If more than one direction of incidence is to be examined, then this value is the increment in the direction of incidence in the φ direction.

EIR8
: Axial ratio v of the polarisation ellipse in the range $-1 \ldots 1$ with the following meaning:
$v = -1$: LHC (left hand circular) polarised incident wave.
$-1 < v < 0$: Elliptical polarisation with axial ratio $|v|$, left hand rotating.
$v = 0$: Linear polarisation.
$0 < v < 1$: Elliptical polarisation with axial ratio $|v|$, right hand rotating.
$v = 1$: RHC (right hand circular) polarised incident wave.

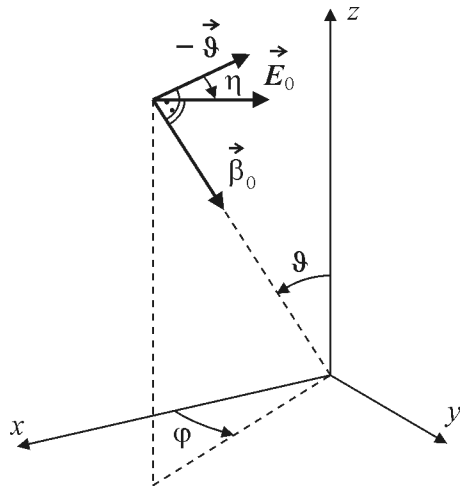


Figure 9-3: Direction of incidence $\vec{\beta}_0$ and direction of polarisation η of the incident field.

The direction of incidence $\hat{\beta}_0$ is specified by the incidence angles ϑ and φ. The polarisation angle η (measured from the negative of the spherical coordinate system unit vector $\hat{\vartheta}$) and the field strength vector $\vec{E}_0$ is defined as indicated in figure 9-3.

The electric field strength of the incident field is then given by

$$\vec{E}_i(\vec{r}) = \left[\vec{E}_0 + jv(\vec{E}_0 \times \hat{\beta}_0)\right] \cdot e^{-j\vec{\beta}_0\vec{r}} \ .$$

The incident magnetic field is given by

$$\vec{H}_i(\vec{r}) = \frac{1}{Z_F}\,\hat{\beta}_0 \times \vec{E}_i$$

with Z_F the wave impedance in the surrounding free space medium.

It should be noted that the incident power density (which is required, for example, for RCS computations) is given by

$$\vec{S}_i = \frac{1}{2}\,\frac{|\vec{E}_0|^2}{Z_F}\,(1+v^2) \ .$$

It is possible to vary the direction of incidence. This is particularly useful when e.g. determining the monostatic radar cross section. The two parameters EIR3 and EIR4 indicate the direction of the first wave. The direction of incidence is varied in the ϑ direction by the increment DTHEI and in the φ direction by DPHII. In each direction the NTHEI and NPHII angles are examined, in total NTHEI*NPHII incident waves are examined.

If an A0 card with NTHEI*NPHII > 1 is read, then all the following control cards until the next Ax, FR or EN cards will be read into a buffer. All these cards are then processed, in a loop, over all the different angles of incidence. If e.g. the monostatic radar cross section is to be calculated for $\vartheta = 90°$ and $0° \leq \varphi \leq 180°$, the following command is used:

```
...
A0   0     1 181     1.0    0.0   90.0    0.0    0.0    0.0    0.1
FF   2  1  1
EN
```

The FF card is read into the buffer and processed 181 times. Through the use of the parameter FFREQ=2 in the FF card the far field is calculated in the direction of the incident wave.

If more than one direction of incidence is to be examined, the right hand side of the linear equation system is changed, but the matrix remains unchanged. Thus it makes sense, by using the CG card, to use Gauss elimination (default if a CG card is not used) which performs a LU-decomposition of the matrix. When the direction of incidence is varied, then only the relatively fast backward substitution has to be done.

9.2.4 A1 Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
A1		ANFL	ULA				EIR1	EIR2	EIR3	EIR4	EIR5	EIR6		
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

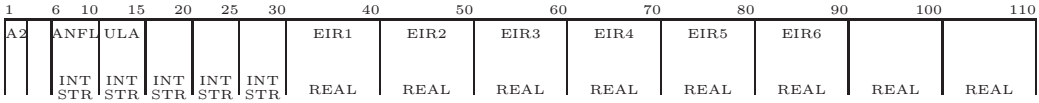
With this card a voltage source is placed on a segment.

Parameters:

ANFL	0: New excitation, replaces all previous excitations. 1: Additional excitation, add to previous excitations.
ULA	Label of the segment, to which the source is applied. If more than one segment has the same label, then source is applied to the last segment with label ULA. Alternatively, one may set ULA=-1, then the feed segment is determined by specifying its Cartesian coordinates x=EIR3, y=EIR4 and z=EIR5.
EIR1	Value of the voltage $\|U_0\|$ in V.
EIR2	Phase of the voltage U_0 in degrees.
EIR3	Only if ULA=-1: the x coordinate of the feed position in m.
EIR4	Only if ULA=-1: the y coordinate of the feed position in m.
EIR5	Only if ULA=-1: the z coordinate of the feed position in m. (The values EIR3, EIR4 and EIR5 are scaled by the SF card if SKALFLAG=1.)
EIR6	The port impedance if this excitation is used in connection with S-parameter calculation. If this field is empty or 0, the value specified at the SP card is used. (This value is only used if the S-parameters are requested with an SP card.)

The vector of the voltage lies in the direction from the beginning of the segment to its end, in the direction in which the segment was created by the BL card. (This is the direction of the current flow through the segment. The internal EMF (electromagnetic force) of the impressed voltage source is in the opposite direction.)

9.2.5 A2 Card



With this card a voltage source is placed at a node between two segments or between a segment and a triangle, ground plane or polygonal plate.

Parameters:

ANFL	0: New excitation, replaces all previous excitations. 1: Additional excitation, add to previous excitations.
ULA	The label of the segment, at which begin point the excitation is located. The excitation has to be located at a node, either between two segments, or between a segment and a triangle, ground plane or polygonal plate. The label must not be ambiguous, i.e. only one segment with this label should be declared. If there is more than one segment with this label then only one node will be fed. Alternatively, one may set ULA=-1, then the feed node is determined by specifying its Cartesian coordinates x=EIR3, y=EIR4 and z=EIR5.
EIR1	Absolute value of the voltage source $\|U_0\|$ in V.
EIR2	Phase of the voltage source U_0 in degrees.
EIR3	Only if ULA=-1: the x coordinate of the feed node in m.
EIR4	Only if ULA=-1: the y coordinate of the feed node in m.
EIR5	Only if ULA=-1: the z coordinate of the feed node in m. (The values EIR3, EIR4 and EIR5 are scaled by the SF card if SKALFLAG=1.)
EIR6	The port impedance if this excitation is used in connection with S-parameter calculation. If this field is empty or 0, the value specified at the SP card is used. (This value is only used if the S-parameters are requested with an SP card.)

There may not be more than two segments connected to the node.

The direction of the vector points in the same direction as the basis function that has been assigned to this node. (When only one segment is connected to the node, the direction is away from the segment.) (This is the direction of the current flow through the node. The internal EMF (electromagnetic force) of the impressed voltage is in the opposite direction.)

9.2.6 A3 Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
A3		ANFL	ULA				EIR1	EIR2	EIR3	EIR4	EIR5	EIR6	EIR7	EIR8
		INT	INT	INT	INT	INT	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL
		STR	STR	STR	STR	STR								

This card realises excitation by a magnetic ring current (TEM-frill) on a segment.

Parameters:

ANFL	0: New excitation, replaces all previous excitations. 1: Additional excitation, add to previous excitations.
ULA	The label of the segment on which the TEM-frill is placed. If there are more than one segments with the same label, the excitation is placed on the last segment. Alternatively, one may set ULA=-1, then the feed segment is determined by its Cartesian coordinates x=EIR5, y=EIR6 and z=EIR7.
EIR1	Absolute value of the voltage $\|U_0\|$ in V.
EIR2	Phase of the voltage U_0 in degrees.
EIR3	Radius of the inner conductor of the coaxial feed.
EIR4	Radius of the outer conductor of the coaxial feed.
EIR5	Only if ULA=-1: the x coordinate of the feed position in m.
EIR6	Only if ULA=-1: the y coordinate of the feed position in m.
EIR7	Only if ULA=-1: the z coordinate of the feed position in m. (The values EIR5, EIR6 and EIR7 are scaled by the SF card if SKALFLAG=1.)
EIR8	The port impedance if this excitation is used in an S-parameter calculation. If this field is empty or 0, the value specified at the SP card is used. (This value is ignored if no SP card is used.)

The vector of the excitation points in the direction from the start point to the end point of the segment, i.e. in the direction in which the segment was created in the BL card. The excitation is not, as in the previously mentioned card, an impressed electric field strength, but it is a magnetic ring current.

As a rule of thumb, it can be said that the radius of the inner conductor must be the same as the radius of the segment and that the outer radius should be 2 to 3 times the size of the inner. If an impedance Z is desired, then the following relation can be used

$$Z \approx 60\,\Omega \cdot ln\frac{\text{EIR4}}{\text{EIR3}}$$

to determine the outer radius. For $Z = 50\,\Omega$ EIR4 should be equal to 2.3*EIR3.

9.2.7 A4 Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
A4		ANFL	ULA	EII3			EIR1	EIR2	EIR3	EIR4	EIR5	EIR6	EIR7	
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card creates a coaxial attachment feed approximation for use in connection with the Green's function for planar substrates with a metallic ground plane (GF card, section 9.2.27).

Parameters:

ANFL
: 0: New excitation, replaces all previous excitations.
: 1: Additional excitation, add to previous excitations.

ULA
: The label of the triangle to excite. The feed point is at the centroid of the triangle (see also figure 9-4). If there are more than one triangle with this label, the excitation is placed on the one with the highest element number. Alternatively, the user may set ULA=-1, and specify the Cartesian coordinates x=EIR3, y=EIR4 and z=EIR5 of the feed point. FEKO will then excite the triangle whose centroid is closest to the specified coordinate.

EII3
: 0: No correction of the impedance, it is computed directly at the excitation point.
: 1: Use an inductive approximation of the feed pin, and transform the impedance such that it is referenced to the ground plane.

EIR1
: Absolute value of the excitation current $|\vec{I}_0|$ in A. The positive current direction is the positive z direction.

EIR2
: Phase of the current $\vec{I}_0$ in degrees.

EIR3
: Only if ULA=-1: the x coordinate of the feed position in m.

EIR4
: Only if ULA=-1: the y coordinate of the feed position in m.

EIR5
: Only if ULA=-1: the z coordinate of the feed position in m.

EIR6
: Only if EII3=1: the radius of the coaxial probe feed pin in m.
: (The values EIR3, EIR4, EIR5 and EIR6 are scaled by the SF card if SKALFLAG=1.)

EIR7
: The port impedance if this excitation is used in connection with S-parameter calculation. If this field is empty or 0, the value specified at the SP card is used. (This value is only used if the S-parameters are requested with an SP card.)

The excitation is shown in figure 9-4. A typical application of the A4 card is given in `example_30a.pre` (*Examples Guide*). It is of course possible to discretise the vertical pin (into segments) and feed one of the segments with a voltage source (A1 card). The advantage of the A4 card is that there are no vertical currents, which results in substantially simpler Green's functions and a significant reduction in computing time.

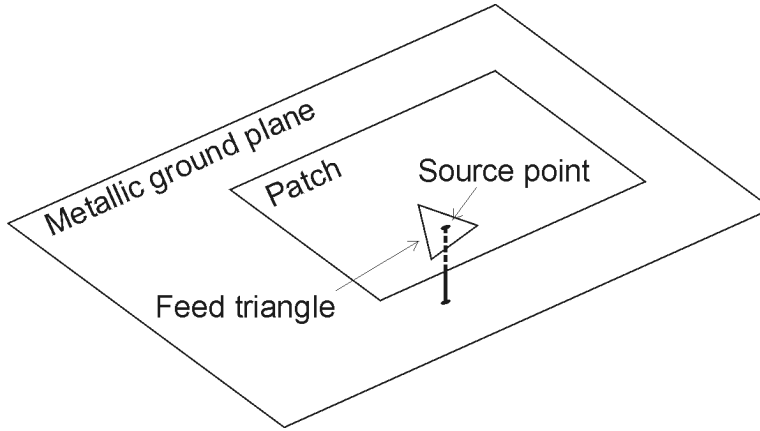


Figure 9-4: Excitation of a patch antenna with a vertical pin

9.2.8 A5 Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
A5	ANFL						EIR1	EIR2	EIR3	EIR4	EIR5	EIR6	EIR7	
	INT	INT	INT	INT	INT		REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL
	STR	STR	STR	STR	STR									

This card specifies excitation by an electric Hertzian dipole.

Parameters:

ANFL	0: New excitation, replaces all previous excitations. 1: Additional excitation, add to previous excitations.
EIR1	Absolute value of the complex amplitude $I \cdot l$ in Am.
EIR2	Phase of the complex amplitude $I \cdot l$ in degrees.
EIR3	x coordinate of the position of the dipole in m.
EIR4	y coordinate of the position of the dipole in m.
EIR5	z coordinate of the position of the dipole in m. (The values EIR3, EIR4 and EIR5 are scaled by the SF card if SKALFLAG=1.)
EIR6	Orientation of the dipole in space: Angle ϑ in degrees from the z axis, analogue to the incidence direction in figure 9-3.
EIR7	Orientation of the dipole in space: Angle φ in degrees is the projection of the dipole onto the plane $z = 0$ opposite the x axis, analogue to the angle of incidence in figure 9-3.

The dipole moment of the electric dipole is given by

$$p = \frac{I\,l}{j\omega}$$

The power radiated by the dipole in a free space environment is given by

$$P = \frac{\beta_0^2 \cdot Z_{F0} \cdot |I\,l|^2}{12\pi} = \frac{\omega^2 \mu_0^2 |I\,l|^2}{12\pi Z_{F0}}$$

FEKO, however, considers the properties of the medium in which the dipole is located as well as the coupling of the dipole with surrounding structures or other sources (for example other Hertzian dipoles in an array antenna) when calculating the power radiated by the Hertzian dipole.

9.2.9 A6 Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
A6		ANFL				EII5	EIR1	EIR2	EIR3	EIR4	EIR5	EIR6	EIR7	
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card specifies excitation by an elementary magnetic dipole.

Parameters:

ANFL
- 0: New excitation, replaces all previous excitations.
- 1: Additional excitation, add to previous excitations.

EII5
- 0: Use the model of an electric ring current for the magnetic dipole (loop current I, enclosed surface A).
- 1: Use the model of a magnetic current element (magnetic line current I_m, length l).

EIR1 Absolute value of the complex amplitude: $I \cdot A$ in Am^2 for EII5 = 0, or $I_m \cdot l$ in Vm for EII5 = 1.

EIR2 Phase of the complex amplitude ($I \cdot A$ or $I_m \cdot l$) in degrees.

EIR3 x coordinate of the dipole position in m.

EIR4 y coordinate of the dipole position in m.

EIR5 z coordinate of the dipole position in m.
(The values EIR3, EIR4 and EIR5 are scaled by the SF card if SKALFLAG=1.)

EIR6 Orientation of the dipole in space: Angle ϑ in degrees from the z axis, analogue to the incidence direction in figure 9-3.

EIR7 Orientation of the dipole in space: Angle φ in degrees is the projection of the dipole onto the plane $z = 0$ opposite the x axis, analogue to the angle of incidence in figure 9-3.

The dipole moment depends on EII5 and is given by

$$m = \frac{I_m\, l}{j\omega} = \mu\, I\, A$$

The power radiated by the dipole in a free space environment is given by

$$P = \frac{\beta_0^4 \cdot Z_{F0} \cdot |I\, A|^2}{12\pi}$$

FEKO, however, considers the properties of the medium in which the dipole is located as well as the coupling of the dipole with surrounding structures or other sources (for

example other magnetic dipoles in an aperture approximation — see the AP card) when calculating the power radiated by the Hertzian dipole.

Even though the two formulations EII5=0 (electric ring current) and EII5=1 (magnetic dipole) result in the same near and far fields if the dipole moment m is the same, the radiated potentials are different. The electric ring current model gives rise to a magnetic vector potential $\vec{A}$, while the magnetic dipole model results in an electric vector potential $\vec{F}$ as well a magnetic scalar potential Ψ.

9.2.10 A7 Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
A7		ANFL	ULA				EIR1	EIR2	EIR3	EIR4	EIR5			
		INT	INT	INT	INT	INT	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL
		STR	STR	STR	STR	STR								

This card is used to specify a voltage source on an edge between two triangles or at a connection edge between a single triangle and a PEC ground plane or UTD plate. As it is substantially simpler, it is strongly recommended to use the AE card rather than the A7 card. (The A7 card is supported only for compatibility with FEKO input files that were created before the AE card became available.)

Parameters:

ANFL 0: New excitation, replaces all previous excitations.
1: Additional excitation, add to previous excitations.

ULA A triangle with the label ULA is searched for. The excitation is placed on the edge that lies opposite to the first corner of the triangle. Once again the label must be unambiguous, i.e. if possible only one triangle must have this label. If there is more than one triangle with this label then only one will be fed. Alternatively, one may set ULA=-1, then the feed edge is determined by specifying its Cartesian coordinates x=EIR3, y=EIR4 and z=EIR5. The edge must be an internal edge, i.e. it must not lie on the edge of a surface (except when connected to a PEC ground plane or UTD plate).

EIR1 Absolute value of the voltage $|U_0|$ in V.

EIR2 Phase of the voltage U_0 in degrees.

EIR3 If ULA=-1: the x coordinate of the edge centre (in m).

EIR4 If ULA=-1: the y coordinate of the edge centre (in m).

EIR5 If ULA=-1: the z coordinate of the edge centre (in m).
(The values EIR3, EIR4 and EIR5 are scaled by the SF card if SKALFLAG=1.)

If two triangles are connected to the edge, the basis function between these triangles is excited. The vector direction of the voltage source lies in the same direction as the basis function associated with this edge. (This is the direction of the current flow through the edge. The internal EMF (electromagnetic force) of the impressed voltage source is in the opposite direction.)

In certain special cases there may be only one triangle connected to the edge. If the edge lies in the plane of a polygonal UTD plate or a PEC ground plane (specified with a GF or BO card), the excitation is placed on the appropriate basis function connecting the triangle to the plate/plane. The positive feed direction is then towards the edge.

9.2.11 AC Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
AC		I_1	I_2	I_3	I_4	I_5	R_1	R_2				"Filename"		
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card inputs data from a `*.rsd` file containing the geometry of a transmission line and the current distribution along this line for one or more frequencies. Such a `*.rsd` file is created, for example, by the transmission line simulation program CableMod[10] or with the OS card in FEKO. The excitation is due to the electromagnetic fields radiated by these line currents (the CM card allows the treatment of electromagnetic fields coupling into lines).

Parameters:

I_1 0: New excitation, replaces all previous excitations.
1: Additional excitation, add to previous excitations.
If the imported `*.rsd` file contains currents for several frequencies, I_1 must be set to 0 as the AC card then results in a frequency loop and currents with different frequencies cannot be superimposed. (If it is not 0, PREFEKO will give an error).

I_2 0: No execution, do not read the `*.rsd` file — this option is used to specify the end of the frequency loop, see below.
1: The line geometry, frequency and currents are read from the `*.rsd` file, and the line is modelled with an array of Hertzian dipoles (see the A5 card). The number of dipoles per line segment is specified with the parameter I_3. Note that this model is only valid if the line segments do not make electrical contact with any conducting surface. (All the segments in the `*.rsd` file must be of the type *intern* and not *loaded*.)
2: The line geometry, frequency and currents are read from the `*.rsd` file, and the line is modelled with a continuous current distribution using one AI card per line segment. (The AI cards are created automatically by PREFEKO when importing the `*.rsd` file.) If a line segment has a *loaded* endpoint it is automatically modelled by an AV card to allow the electrical contact.

I_3 In the case $I_2 = 1$, the parameter I_3 specifies the number of Hertzian dipoles per line segment.

[10] To use the CableMod interface this module must be activated, if required please contact EMSS.

I_4 0: Use the discrete frequency values as they are specified in the `*.rsd` file.

1: Only read the minimum and maximum frequency from the `*.rsd` file and obtain a continuous solution in this frequency band using adaptive frequency sampling. If this flag is used, only one AC card (and no FR cards) is permitted in the `*.pre` file.

I_5 Only used for $I_4 = 1$, then it specifies the maximum number of discrete frequency points to be used during the adaptive analysis (see also the parameter NFREQ at the FR card for FREQF=2).

R_1 In the case $I_2 = 2$, the parameter R_1 specifies the radius of the impressed current elements. This parameter is optional and is passed on to the AI and AV cards. (See the description of EIR6_2 in these cards.) If R_1 is zero or empty a current filament approximation is used.

R_2 Only used for $I_4 = 1$, then it specifies the minimum frequency stepping (see also the parameter DFREQ at the FR-card for FREQF=2).

"Filename" The name of the `*.rsd` file to read. The filename must be enclosed in quotation marks and start in column 81 (or after that) of the `*.pre` file.

The frequency is defined in the `*.rsd` file, thus the preceding FR cards are ignored when processing an AC card.

All commands after the AC card in the FEKO input file (for example FF, FE, OS, GF, BO, ...) are processed within a frequency loop through all the frequencies in the `*.rsd` file. The loop is terminated by any of the following three cards (these cards are not included in the loop they terminate)

- AC (importing a new `*.rsd` file, or using the flag $I_2 = 0$)
- FR (manually setting a new frequency)
- EN (end of the FEKO input file)

For example, if a CableMod file must be read and the near field calculated for each frequency, the input file may look as follows

```
AC  ...      ** Read the *.rsd file
FE  ...      ** Calculate the near field
EN           ** End
```

However, if one wants to analyse, for example, a metal plate, which is excited first by an impressed line current and then also by a plane wave (in each case the near fields and the currents on the plate must be written to the output file), the input file would be

```
** Excitation by a line current
AC  ...      ** Read the *.rsd file
FE  ...      ** Calculate the near field
OS  ...      ** Output the currents

** Excitation by a plane wave
FR  ...      ** Set the frequency and terminate AC loop
AO  ...      ** Specify plane wave excitation
FE  ...      ** Calculate the near field
OS  ...      ** Output the currents

** End
EN
```

9.2.12 AE Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
AE	I_1	I_2	I_3	I_4		EIR1	EIR2	EIR3						
	INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL	

This card specifies an excitation at an edge between triangular surface elements similar to the A7 card. The AE card, however, has the advantage that the location of the feed point and the positive feed direction are substantially easier to specify. In addition it is possible to specify a feed edge which contains a number of triangle edges as shown in figure 9-5.

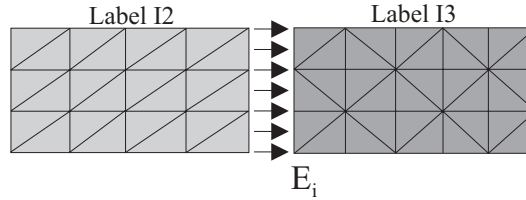


Figure 9-5: Example of the use of the AE card

Parameters:

I_1 0: New excitation, replaces all previous excitations.
1: Additional excitation, add to previous excitations.

I_2/I_3 The meaning of these parameters depends on I_4.

I_4 0: The excitation is placed on the edge between the regions with labels I_2 and I_3. The positive source direction is from the side with label I_2 to the one with label I_3.
2: Excite the edges of metallic triangles with label I_2 which are connected to UTD surfaces or to a PEC ground plane (as specified with a BO or GF card). I_3 is not used. The positive feed direction is towards the UTD region or ground plane.
3: Special microstrip line feed. The excitation is placed on all edges on the line between points (previously specified with DP cards) I_2 and I_3. A GF card with a conducting ground plane must be present. The positive source direction is from the triangles towards the edge.

EIR1 Absolute value of the voltage $|U_0|$ in V.

EIR2 Phase of the voltage U_0 in degrees.

EIR3 The port impedance if this excitation is used in connection with S-parameter calculation. If this field is empty or 0, the value specified at the SP card is used. (This value is only used if the S-parameters are requested with an SP card.)

The positive source direction as used above is the direction of the current flow through the edge. The internal EMF (electromagnetic force) of the impressed voltage source is in the opposite direction.

It should be noted that the edge between the surfaces with labels I_2 and I_3 (for $I_4 = 0$) does not have to be straight. One may, for example, excite two half cylinders against each other. If an impedance must also be applied to the edge, the AE card can be combined with the LE card.

For $I_4 = 0$, more than two triangles may be connected to each edge section as shown in figure 9-6. However, one of these triangles must have a unique label and all the other triangles connected to this edge must have the same label. In figure 9-6 there is one triangle with label 2 at each edge between triangles and all the others have label 1. One may therefore specify an edge between labels 1 and 2. The edge in figure 9-7 cannot be used as a feed edge as there are more than one triangle of each of the two labels involved. The edge shown in figure 9-8 cannot be used as a feed edge either as there are triangles with three different labels connected to each edge.

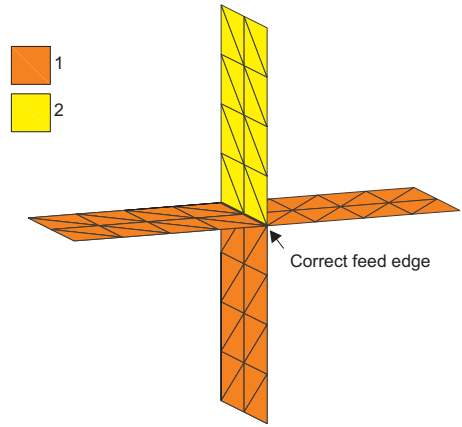


Figure 9-6: Example of a feed edge where more than two triangles are connected

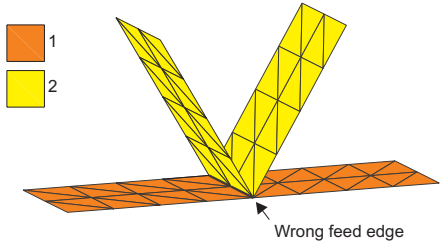


Figure 9-7: Disallowed feed edge (there is no triangle with a unique label at the edge)

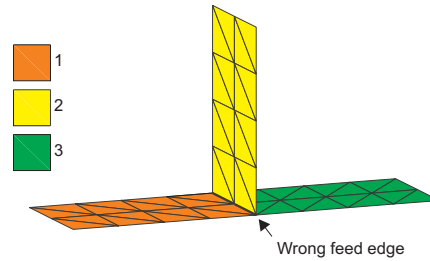


Figure 9-8: Disallowed feed edge (triangles with three different labels join at the edge)

9.2.13 AI Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
AI		ANFL					EIR1 EIR1_2	EIR2 EIR2_2	EIR3 EIR3_2	EIR4 EIR4_2	EIR5 EIR5_2	EIR6_2		
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this *two line* card an impressed current source is specified. The current varies linearly between the value at the start point and that at the end point, see figure 9-9.

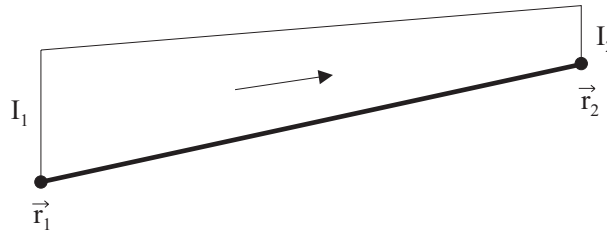


Figure 9-9: Impressed line current with a linear current distribution

Parameters:

ANFL	0: New excitation, replaces all previous excitations. 1: Additional excitation, add to previous excitations.
EIR1	Amplitude $\|I_1\|$ in A of the current at the start point $\vec{r}_1$.
EIR2	Phase of the current at the start point in degrees.
EIR3	x coordinate of the start point $\vec{r}_1$ in m (Note that all the coordinate values are scaled by the SF card if SKALFLAG=1.)
EIR4	y coordinate of the start point $\vec{r}_1$ in m.
EIR5	z coordinate of the start point $\vec{r}_1$ in m.
EIR1_2	Amplitude $\|I_2\|$ in A of the current at the end point $\vec{r}_2$.
EIR2_2	Phase of the current at the end point in degrees.
EIR3_2	x coordinate of the end point $\vec{r}_2$ in m (see comment for EIR1).
EIR4_2	y coordinate of the end point $\vec{r}_2$ in m.
EIR5_2	z coordinate of the end point $\vec{r}_2$ in m.
EIR6_2	This parameter is optional. If specified, and different from zero, this value gives a finite wire radius for the impressed current element. FEKO then assumes that the current is uniformly distributed on the wire surface and uses the exact wire integral. If the parameter EIR6_2 is not specified, the current filament approximation is used. (This value is scaled by the SF card if SKALFLAG=1.)

The following restrictions apply when using the impressed current elements.

- It is not possible to attach the impressed current to a wire segment in the FEKO model. (If the impressed current is making electrical contact with a triangular surface current element, the AV card should be used.)
- When modelling dielectric bodies with the surface equivalence method, the current element must be in the free space medium, i.e. outside the dielectric bodies. (The material parameters of this medium can, however, be set with the EG and/or GF cards).
- When used with the spherical Green's function, the current element must be outside the dielectric spheres.
- The current segments may be joined with each other and with the AV card to form long paths and/or closed loops. The point charges which arise when the current does not go to zero at an end point or when there is a current discontinuity at a connection point, are not taken into consideration. This is required to model, for example, the case where radiating lines are terminated in a non-radiating structure. If these charges must be considered explicitly, the line current should be modelled by a row of Hertzian dipoles (see the A5 card). Note, however, that the constant line charge along the current segment is correctly taken into account.
- If several of these current elements are used, the total radiated power (required to calculate, for example, the far field gain/directivity) can only be calculated accurately if the mutual coupling between segments is taken into account. Due to neglecting the point charges at the ends of the segments, the coupling cannot be determined accurately. If exact values of the radiated power are required, it should be determined by integrating the far field (see the FF card). It should be noted that, for example, the computed near and far fields (the actual field strength values), the induced currents, coupling factors, losses are computed correctly.

9.2.14 AP Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
AP		ANFL	TYPE	S_1	S_2	S_3	NSTART	N2	N3	AMP	PHASE	"Filename"	"Filename"	
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card a planar, cylindrical or spherical aperture of measured or calculated field values is converted into an equivalent array of electric and magnetic dipoles. The card is processed by PREFEKO and replaced by the required number of A5 and A6 cards in the `*.fek` file.

Parameters:

ANFL
- 0: New excitation, replaces all previous excitations.
- 1: Additional excitation, add to previous excitations.

TYPE
- ±1: Read electric field values for a planar aperture from a file, in `*.efe` format.
- ±2: Read magnetic field values for a planar aperture from a file, in `*.hfe` format.
- ±3: Read electric field values for a planar aperture from a text file (see below).
- ±4: Read magnetic field values for a planar aperture from a text file (see below).
- ±5: Read both electric and magnetic field values for a planar aperture from files, in `*.efe` and `*.hfe` format respectively.
- ±6: Read both electric and magnetic field values for a planar aperture from text files (see below).
- ±7: Electric field values for a planar aperture follow in the `*.pre` input file (see below).
- ±8: Magnetic field values for a planar aperture follow in the `*.pre` input file (see below).
- ±9: Both electric and magnetic field values for a planar aperture follow in the `*.pre` input file (see below).
- ±15, ±16, ±19: The same as ±5, ±6, ±9 but for a cylindrical aperture rather than a planar one. Note that in this case both the electric and magnetic fields are required.
- ±25, ±26, ±29: The same as ±5, ±6, ±9 but for a spherical aperture rather than a planar one. Note that in this case both the electric and magnetic fields are required.

The sign of the parameter TYPE is used to determine if dipoles should lie on the edges of the aperture or not (see figures 9-10 to 9-13). For positive values of TYPE the dipoles lie on the edges, for negative values the dipoles lie half an increment away from the respective edges. When two apertures have a common side, dipoles should not lie on the edges of both apertures otherwise two dipoles may have the same location and polarisation. If this is the case the power calculation in FEKO will fail.

S_1 The name of a node point that defines the origin of the aperture.

S_2, S_3 For a planar aperture, S_2 and S_3 specify the names of two nodes that define the corners of the aperture in the $\hat{u}_2$ and $\hat{u}_3$ directions respectively (see figure 9-10). For cylindrical and spherical apertures they specify the directions of the local z and x axes respectively (see figures 9-12 and 9-13).

NSTART The number of the first field point to be used for the aperture. For NSTART=1 field values are read from the start of the file, for larger values the first NSTART-1 values (`*.efe` and `*.hfe` files) or lines (text files) are ignored. This may be used, for example, if the data file contains the field values for more than one aperture. NSTART is not used if the field data is obtained from the `*.pre` input file.

N2 The number of field points in the S_1–S_2 (planar aperture), $\hat{\varphi}$ (cylindrical aperture) or $\hat{\vartheta}$ (spherical aperture) direction.

N3 The number of field points in the S_1–S_3 (planar aperture), $\hat{z}$ (cylindrical aperture) or $\hat{\varphi}$ (spherical aperture) direction.

AMP A constant by which the amplitudes of all the dipoles in the aperture are scaled.

PHASE A constant phase added to all dipoles in the aperture.

"Filename" The input filename (`*.efe`, `*.hfe` or text file), from which the field data must be read. The filename must be enclosed in double quotation marks and must be entered at or after column 81 in the `*.pre` file.

"Filename" When both electric and magnetic fields are read from file, a second filename must be specified for the magnetic field data. This follows (also within double quotation marks) at an arbitrary location after the first filename.

The aperture is based on the equivalence principle. This states that the sources and scatterers inside a given volume can be removed, and modelled by placing the equivalent currents $\vec{J}_s = \hat{n} \times \vec{H}$ and $\vec{M}_s = -\hat{n} \times \vec{E}$ on the enclosing surface. The vector $\hat{n}$ is a unit vector, normal to the surface, and points towards the exterior region. The fields in this region are the same as the original fields, while those in interior region are zero.

Field values are read from the data files (with a possible offset specified with NSTART) or the `*.pre` input file and converted to equivalent electric (magnetic fields) and magnetic (electric field) dipoles at these points. *Note that all angles are read from the data, but no distance values.* Thus for planar apertures the positions are calculated entirely from the specified points (S_1, S_2 and S_3). For cylindrical apertures S_1 and S_2 specify the extents of the aperture along the local $\hat{z}$ direction and S_1–S_3 specifies the direction of the x axis as well as the radius of the cylinder. The points are placed at the φ-values as specified with the field data. For spherical apertures, S_1–S_2 specifies the direction of the z axis and S_1–S_3 the x axis. S_2 and S_3 must lie on the same radius which is also the radius of the field points. In this case both ϑ and φ are read with the data.

The dipole amplitude is the product of the surface current and the incremental area between samples. In addition, the amplitude of the dipoles on the sides (only for positive values of TYPE) are reduced by a factor of 2 and those on the corners by a factor of 4 such that the effective aperture of integration has the same size as the specified aperture.

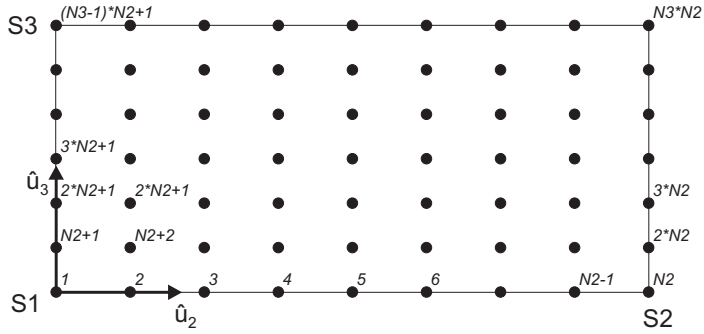


Figure 9-10: Location of the equivalent dipoles on a planar aperture with positive TYPE.

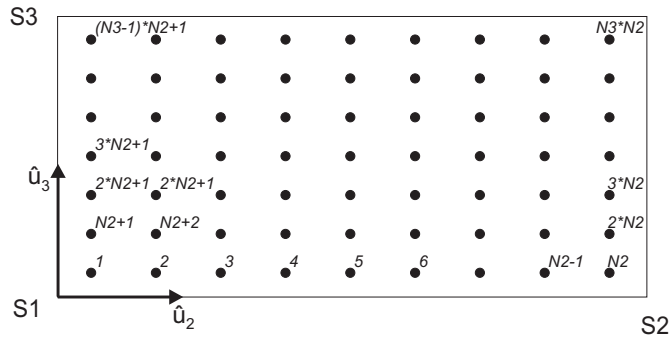


Figure 9-11: Location of the equivalent dipoles on a planar aperture with negative TYPE.

Figures 9-10 and 9-11 show the application of the equivalence principle to a planar aperture. There are respectively N2 and N3 field points along the two orthogonal directions. For positive values of the parameter TYPE the first point lies at S_1 with the following points in the direction of S_2 as shown by the indices in figure 9-10. For negative values of TYPE the pattern is as shown in figure 9-11. The normal vector is calculated from $\hat{n} = \hat{u}_2 \times \hat{u}_3$ with $\hat{u}_2$ and $\hat{u}_3$ as defined in the figure.

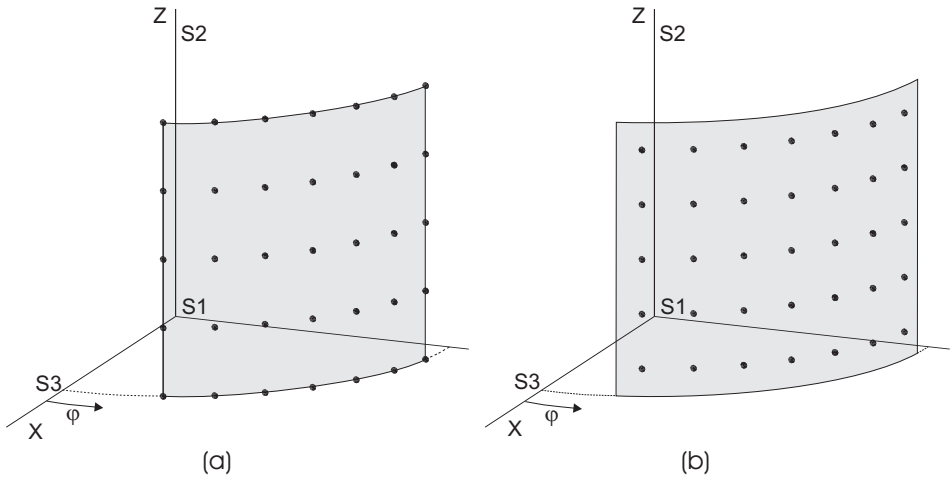


Figure 9-12: Location of the equivalent dipoles on a cylindrical aperture.
(a) Positive TYPE. (b) Negative TYPE.

For cylindrical and spherical apertures PREFEKO will determine which coordinate is incremented first and write out the dipoles accordingly.

Figure 9-12 shows the dipole locations for a cylindrical aperture created from a data file containing field values for φ from 20° to 80° in 10° increments and 5 values in the z direction. When TYPE is positive such that samples lies up to the edges of the aperture, the points and the effective aperture is a shown in figure (a). When TYPE is negative, samples does not lie on the edges as shown in figure (b). Note that, when using identical input data as for positive values of TYPE, the z positions of the samples changed, while in the φ direction the size of the effective aperture is increased by 5° on both sides.

Figure 9-13 shows the dipole locations for a spherical aperture created from field values for ϑ from 40° to 80° with 10° increments and φ from 20° to 80° also with 10° increments. In this case the aperture increases in size in both directions when TYPE is negative.

A fully closed surface can be created by specifying 6 planar apertures or a spherical one. The surface equivalence principle can be applied to this surface by reading both electric and magnetic fields for each plane. (For planar apertures the user should specify 6 AP cards which each use both electric and magnetic fields. If separate cards are used for the

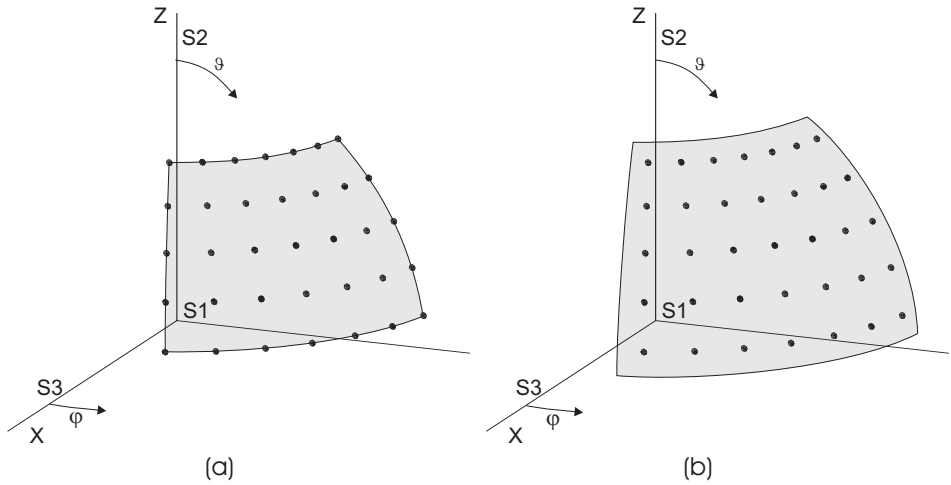


Figure 9-13: Location of the equivalent dipoles on a spherical aperture.
(a) Positive TYPE. (b) Negative TYPE.

electric and magnetic fields the radiated power is not calculated correctly.) The normal vector must point to the exterior region, normally this is outward. (For planar apertures created form `*.efe` and `*.hfe` files, the sample order determines the directions of $\hat{u}_2$ and $\hat{u}_3$ which, in turn, determines the normal vector $\hat{n} = \hat{u}_2 \times \hat{u}_3$. If this is pointing into the cube, an additional 180° phase shift is obtained by setting PHASE=180. This change the sign of the field radiated by the aperture which, when interacting with the remaining sources, will result in the correct total fields in the desired region.) All surfaces and scatterers inside the body must be removed and those outside retained.

For planar apertures (for example, the opening of a horn antenna), one may use the mirror principle if the field at the edges can be neglected. This results in a duplication of the magnetic current and cancellation of the electric current. Thus it is sufficient to read only the electric fields and scale by the factor AMP=2. In this case any sources or structures in the region towards which the normal is pointing, should also be subjected to the mirroring (i.e. the structures should be electrically mirrored by using the SY card). Further, it should be remembered that the fields will only be correct in the direction that the normal vector points to. The symmetric fields in the other half-space will not be equal to the fields of the original problem. Note that FEKO takes this into account and divides the total radiated power by two when calculating the power radiated by a planar aperture containing only electric or magnetic fields.

The parameter TYPE is also used to determine the format of the data file. For TYPE=$\pm$1, $\pm$2, $\pm$5, $\pm$15 and $\pm$25 the fields are calculated in FEKO (FE card) and written to `*.efe` and `*.hfe` files on request of the DA card. Note again that the position (distance) data in the files is not used, but the angle information is used.

For TYPE=±3, ±4, ±6, ±16 or ±26 the data is read from an ASCII format text file. Each line in the file represents one point and the values are space delimited. For planar apertures it must have four parameters: The absolute value and phase (in degrees) of the field component in the $\hat{u}_2$ direction followed by the absolute value and phase in the $\hat{u}_3$ direction (see the example below). The data must be such that the position increments along the $\hat{u}_2$ direction first. For cylindrical apertures it must have five parameters: The angle φ (in degrees) followed by the absolute value and phase of the $\hat{\varphi}$ component and the absolute value and phase of the $\hat{z}$ component. For spherical apertures it must have six parameters: The angles ϑ and φ followed by the absolute value and phase of the $\hat{\vartheta}$ and $\hat{\varphi}$ components.

For TYPE=±7, ±8, ±9, ±19 or ±29 the data is read from the `*.pre` input file itself. In this case the data must be in the normal column based input format and FOR loops etc. may be used. The four field components are the same as for the text data, and must be entered in the real fields R_3 to R_6 — i.e. columns 51 to 90. The angle φ occurs in R_2 and ϑ in R_1 when they are required. If both electric and magnetic fields are required, all N2*N3 electric fields are given first followed by the same number of magnetic fields.

Example of AP card usage:

As an example consider an open ended X-band waveguide radiating through a hole in a large ground plane as shown in figure 9-14. Away from the aperture the plane $z = 0$ is perfectly conducting, i.e. the tangential electric field is zero, while the magnetic field is not — thus we will use electric symmetry.

For this example the field is considered to be purely y directed (i.e. it has only a $\hat{y}$, or $\hat{u}_3$, component). The field is assumed to be constant in the y direction and to have a cosine distribution in the x direction (i.e. the $\hat{u}_2$ axis).

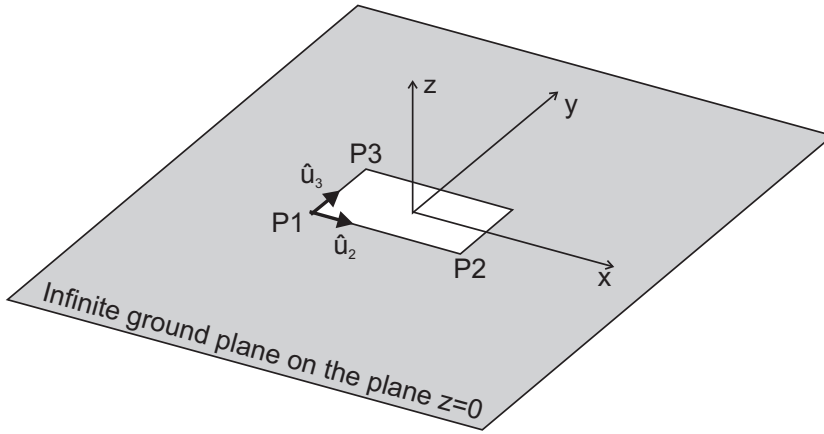


Figure 9-14: Example of an open waveguide as an implementation of the AP card

With N2=5 and N3=3 — in practice more points may be required — the data file will be as follows:

```
   0.0    0.0    0.0    0.0
   0.0    0.0    0.707  0.0
   0.0    0.0    1.0    0.0
   0.0    0.0    0.707  0.0
   0.0    0.0    0.0    0.0
   0.0    0.0    0.0    0.0
   0.0    0.0    0.707  0.0
   0.0    0.0    1.0    0.0
   0.0    0.0    0.707  0.0
   0.0    0.0    0.0    0.0
   0.0    0.0    0.0    0.0
   0.0    0.0    0.707  0.0
   0.0    0.0    1.0    0.0
   0.0    0.0    0.707  0.0
   0.0    0.0    0.0    0.0
```

The zero values will not result in any dipoles, but they must be in the data file to allow correct indexing. The `*.pre` file will contain the following section:

```
...

** Only electric fields --- use electric symmetry in the z=0 plane
SY   1    0    0    2

** Define the corner points of the aperture
#wx = 0.02286
#wy = 0.01016
DP     P1                    -#wx/2    -#wy/2     0
DP     P2                     #wx/2    -#wy/2     0
DP     P3                    -#wx/2     #wy/2     0

** The geometry ends after the corner nodes have been defined
EG   1    0    0    0    0

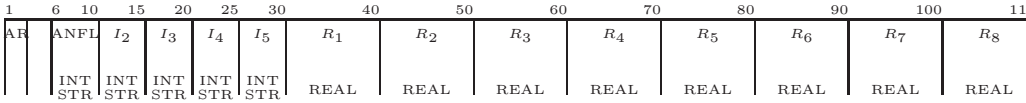
** Specify the frequency
FR   1    0                  9.375e9

** Specify the AP card as a new source
** The amplitude factor of 2.0 is due to the use of the equivalence principle
AP   0    3    P1   P2   P3   1        5        3        2.0      0.0    "Guide.dat"

...
```

which will generate nine x directed magnetic dipoles of the correct magnitude in the `*.fek` file.

9.2.15 AR Card



With this card the radiation pattern of an antenna is used as an impressed source. The pattern is read from a data file or defined in the `*.pre` file below the AR card.

This card has a variety of uses, for example, importing measured radiation patterns, synthesising arrays from the individual patterns of the elements, realising radiation only within certain sectors, etc. In the MoM/UTD hybrid it is possible to simulate, for example, the antenna on its own and to save the far field in a `*.ffe` file. This field is then imported and used as source in the UTD part which may greatly speed up the ray tracing computation as there is now only one source point.

Parameters:

ANFL
- 0: New excitation, replaces all previous excitations.
- 1: Additional excitation, add to previous excitations.

I_2
- 1: Read the radiation pattern from an `*.ffe` format file (which may be created with the DA and FF cards).
- 2: Read the radiation pattern from an ASCII file (the format of this file is described below).
- 3: The radiation pattern is specified in the $I_4 \cdot I_5$ lines following the AR card in the `*.pre` file (the format is described below).

I_3 This parameter is only relevant for $I_2 = 1$ or $I_2 = 2$, and gives the line number of the first line to read from the input file. If the data must be read from the beginning of the file, I_3 should be set equal to 1. This parameter is used when the `*.ffe` file contains more than one pattern. For example, if the file contains the pattern at various frequencies, the correct pattern can be selected by setting the appropriate value of I_3 for each frequency.

I_4 The number of ϑ angles in the pattern.

I_5 The number of φ angles in the pattern.

R_1 The values of the complex field strengths E_ϑ^{FF} and E_φ^{FF} are read from the data file ($I_2 = 1$ or $I_2 = 2$) or the `*.pre` input file ($I_2 = 3$). The parameter R_1 is used to scale the amplitude of the field strength by a constant value.

R_2 The parameter R_2 specifies a constant additional phase for the field strength values.

R_3	x coordinate of the source point (i.e. the position where the antenna is placed) in m, this value is affected by the scale factor of the SF card if SKALFLAG=1.
R_4	y coordinate, similar to R_3 above.
R_5	z coordinate, similar to R_3 above.
R_6	The angle α_x in degrees, with which the imported pattern is rotated around the x axis.
R_7	The angle α_y of the rotation around the y axis, similar to R_6.
R_8	The angle α_z of the rotation around the z axis, similar to R_6.
"Filename"	The name of the `*.ffe` or ASCII input file (for $I_2 = 1$ or $I_2 = 2$). The filename must be enclosed in double quotation marks and entered at or after column 111.

The radiation pattern of the antenna must be specified in spherical coordinates (ϑ, φ) with the phase centre thereof located at the origin of the local (as used in the pattern data) coordinates. If this is not the case, the phase of the far fields will not be correct. (For example, if a `*.ffe` file is exported with FEKO for use with the AR card, the origin should be shifted with the OF card to the phase centre of the antenna.) The vertical and horizontal components of the complex electric field E_ϑ^{FF} and/or E_φ^{FF} must be specified at discrete angles (ϑ_i, φ_j) with $i = 1 \ldots I_4$ and $j = 1 \ldots I_5$. These fields have the unit Volt, the actual far fields are calculated from

$$E_\vartheta = E_\vartheta^{FF} \cdot \frac{e^{-j\beta R}}{R} \quad \text{or} \quad E_\varphi = E_\varphi^{FF} \cdot \frac{e^{-j\beta R}}{R}$$

with R the distance to the field point and β the complex propagation constant in the free space medium (see the EG and GF cards). These formulas are used for all distances R, i.e. also in the near field. However FEKO tests whether the far field conditions are met (by calculating the directivity and equivalent aperture) and gives an appropriate warning if this is not the case.

The permissible range of the angles ϑ_i is $0° \ldots 180°$ and they must be in ascending order, i.e. $\vartheta_{i+1} > \vartheta_i$. However, the angles do not have to be equidistantly spaced. (Thus, for example, for a highly directive antenna, a denser grid can be used close to the main beam direction.) The same applies to the angles φ_j where the permissible range is $0° \ldots 360°$. For field angles outside the start and end values defined in the data (i.e. for $\vartheta < \vartheta_1$, $\vartheta > \vartheta_{I_4}$, $\varphi < \varphi_1$ or $\varphi > \varphi_{I_5}$), the field strengths E_ϑ^{FF} and E_φ^{FF} are set to zero, such that a sector radiator can be realised. The values at field angles within the defined range are determined by bilinear interpolation. To realise a complete radiation pattern, rather than a sector radiator, the angles should be defined such that $\vartheta_1 = 0°$, $\vartheta_{I_4} = 180°$, $\varphi_1 = 0°$ and $\varphi_{I_5} = 360°$.

The format of the data depends on the value of the parameter I_2:

- $I_2 = 1$ (`*.ffe` file)

  With $I_2 = 1$ the radiation pattern is read form lines I_3 to $I_3 + I_4 \cdot I_5 - 1$ of a `*.ffe` file created with FEKO (using the DA and FF cards). All the data of the radiation pattern (angles and field values) are determined from the file. The user should, however, ensure that, for example, the frequency is correct. If an antenna is analysed with FEKO, the far field can, for example, be exported to the `*.ffe` file in 5° angle increments using the commands

  ```
  DA   0    0    1    0    0
  FF   1    37   73   0         0          0          5          5
  ```

  Note that 37 points are used for ϑ and 73 for φ to ensure that the radiation pattern is closed (see also the comment above).

- $I_2 = 2$ (external ASCII file)

  With $I_2 = 2$ the data is read from lines I_3 to $I_3 + I_4 \cdot I_5 - 1$ of the specified external data file. Each line contains 6 space delimited data fields in the following order:

  - The angle ϑ in degrees
  - The angle φ in degrees
  - Amplitude of the field strength E_ϑ^{FF} in V
  - Phase of the field E_ϑ^{FF} in degrees
  - Amplitude of the field strength E_φ^{FF} in V
  - Phase of the field E_φ^{FF} in degrees

  The inner loop should be with respect to the angle ϑ such that the order of the lines is as follows

$$\begin{array}{ccc} \vartheta_1 & \varphi_1 & \cdots \\ \vartheta_2 & \varphi_1 & \cdots \\ \vartheta_3 & \varphi_1 & \cdots \\ \vdots & \vdots & \cdots \\ \vartheta_{I_4} & \varphi_1 & \cdots \\ \vartheta_1 & \varphi_2 & \cdots \\ \vartheta_2 & \varphi_2 & \cdots \\ \vdots & \vdots & \cdots \\ \vartheta_{I_4} & \varphi_{I_5} & \cdots \end{array}$$

- $I_2 = 3$ (pattern is specified in the `*.pre` file)

  The case $I_2 = 3$ is similar to reading an external data file ($I_2 = 2$), except that the data is read directly from the `*.pre` input file. The six data fields mentioned for $I_2 = 2$ must appear in the columns of the parameters R_1 to R_6 in the $I_4 \cdot I_5$ lines following the AR card. The data lines may be separated by comment lines (EditFEKO, however, does not support this) and FOR–NEXT loops may be used. An example is given below.

The radiation pattern, specified in the local spherical coordinate system (ϑ, φ) of the antenna, is read and initially placed at the origin of the global coordinate system in which the `*.pre` file is constructed. The pattern is now rotated by an angle α_z (parameter R_8) around the z axis, by α_y (parameter R_7) around the y axis and by α_x (parameter R_6) around the x axis. (The rotation is identical to the rotation executed by the TG card in section 8.2.35 and the rotation matrix M is applicable to the both the TG and AR cards.) Finally the pattern is shifted to the location specified by the parameters R_3, R_4 and R_5.

If the AR card is used simultaneously with a ground plane (BO card), FEKO automatically includes the influence of the ground plane on the radiation pattern. The imported pattern must therefore be the free space radiation pattern of the antenna (in the absence of the ground plane). If this is not the case the influence of the ground plane is considered twice.

The use of the PW card to specify the radiated power is allowed. The field amplitudes $|E_\vartheta^{FF}|$ and $|E_\varphi^{FF}|$ will be scaled accordingly. Multiple radiation patterns can be used simultaneously, and also with other sources such as an incident plane wave. In such a case, the coupling is not considered when the radiated power is determined.

The AR card cannot be used with special Green's functions for a layered sphere or for a layered substrate.

We conclude this description with a couple of examples.

- Importing an `*.ffe` file

  Above we stated that the commands

```
DA   0    0    1    0    0
FF   1    37   73   0         0.0        0.0        5.0        5.0
```

  create a complete radiation pattern of an antenna in 5° increments. This can then be imported as source into another model with the command

```
AR   0    1    1    37   73   1.0        0.0        0.0        0.0        0.0 ...
                                  ...   0.0        0.0        0.0      "file.ffe"
```

- Sector radiator

  We want to realise an ideal sector radiator, which radiates 10 Watt horizontal polarisation in the angular region defined by $-70° \leq \varphi \leq 70°$ and $75° \leq \vartheta \leq 105°$. Since the angle range of the imported pattern must be positive one may define separate sources for the regions $0° \leq \varphi \leq 70°$ and $290° \leq \varphi \leq 360°$. A more elegant solution is to define a single pattern in the range $0° \leq \varphi \leq 140°$ and rotate it by $-70°$ around the z axis. The complete radiation pattern is be defined in the following input file (note that only horizontal polarisation, i.e. E_{φ}^{FF}, is required):

```
** Application example for the AR card: Sector radiator

** No other structures considered
EG   1   0   0   0   0

** Set the frequency
FR   1   0               100.0e6

** Specified radiated power
PW   1                   10.0

** Define the sector radiator
AR   0   3       2   2   1.0      0.0      0.0      0.0      0.0      0.0      0.0      -70
**                       Theta    Phi       E_theta           E_Phi
                         75       0        0        0        1        0
                         105      0        0        0        1        0
                         75       140      0        0        1        0
                         105      140      0        0        1        0
** Check: Compute the full 3D radiation pattern with 5 deg stepping
FF   1   37  73  0        0.0      0.0      5.0      5.0

** End
EN
```

FEKO determines a directivity of 10.1 dBi. The radiation pattern is easily validated by calculating the far field as shown with the FF card in the last step. The result (as presented by WinFEKO) is shown in figure 9-15.

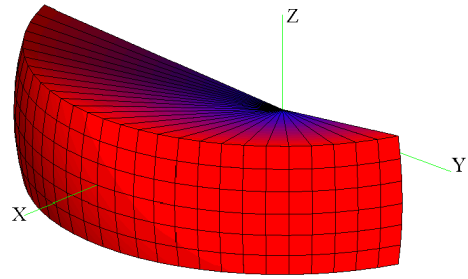


Figure 9-15: 3D radiation pattern of the sector radiatior.

9.2.16 AV Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
AV		ANFL	ULA				EIR1 EIR1_2	EIR2 EIR2_2	EIR3 EIR3_2	EIR4 EIR4_2	EIR5 EIR5_2	EIR6_2		
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this *two line* card an impressed current source is specified similar to that of the AI card, but with the AV card the end point makes electrical contact with a conducting surface as shown in figure 9-16. The current varies linearly between the value at the start point and that at the end point. At the connection point special singular functions are used for the surface current density on the triangles to allow continuous current flow.

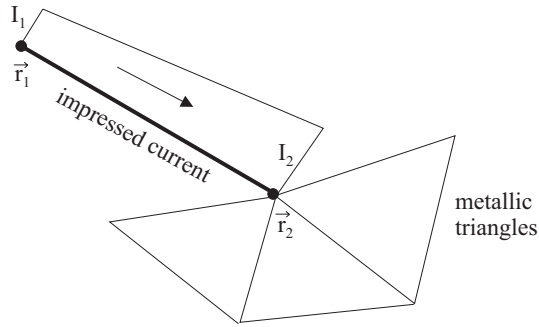


Figure 9-16: Impressed line current with a linear current distribution and electrical contact to conducting triangles

Parameters:

ANFL
- 0: New excitation, replaces all previous excitations.
- 1: Additional excitation, add to previous excitations.

ULA
- 0: The coordinates of the end point $\vec{r}_2$ are known and specified with EIR3_2, EIR4_2 and EIR5_2. This point must coincide with a corner point of one or more triangles.
- 1: The coordinates of the end point $\vec{r}_2$ are not known. In this case the input fields EIR3_2, EIR4_2 and EIR5_2 are not used. FEKO searches through all the metallic triangles for the corner point that is closest to the start point $\vec{r}_1$ of the current element. This is then the end point $\vec{r}_2$.

For both ULA=0 or ULA=1 FEKO automatically searches for all the triangles making electrical contact with the end point.

EIR1 Amplitude $|I_1|$ in A of the current at the start point $\vec{r}_1$.

EIR2 Phase of the current at the start point in degrees.

EIR3	x coordinate of the start point $\vec{r}_1$ in m. (Note that all the coordinate values are scaled by the SF card if SKALFLAG=1.)
EIR4	y coordinate of the start point $\vec{r}_1$ in m.
EIR5	z coordinate of the start point $\vec{r}_1$ in m.
EIR1_2	Amplitude $\|I_2\|$ in A of the current at the end point $\vec{r}_2$.
EIR2_2	Phase of the current at the end point in degrees.
EIR3_2	For ULA = 0: x coordinate of the end point $\vec{r}_2$ in m (see comment for EIR1).
EIR4_2	For ULA = 0: y coordinate of the end point $\vec{r}_2$ in m.
EIR5_2	For ULA = 0: z coordinate of the end point $\vec{r}_2$ in m.
EIR6_2	This parameter is optional. If specified, and different from zero, this value gives a finite wire radius for the impressed current element. FEKO then assumes that the current is uniformly distributed on the wire surface and uses the exact wire integral. If the parameter EIR6_2 is not specified, the current filament approximation is used. (This value is scaled by the SF card if SKALFLAG=1.)

The following restrictions apply when using the impressed current elements making electrical contact with conducting surfaces.

- All the restrictions given in the discussion of the AI card also apply in this case.
- The start point of the impressed current segment may be connected with AI cards or further AV cards. If there is a current discontinuity at this point, the resulting point charge is not considered (see the discussion given with the AI card). Line charges along the current path and surface charges on the triangles are correctly taken into account. At the connection point $\vec{r}_2$ a continuous current model is used such that a point charge is not possible here.

9.2.17 BO Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
BO		FLAGRF					EPSRF	SIGRF	MUERF	TAND MUERF	TAND EPSRF			
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card a ground plane (at $z = 0$) can be specified for all computations following the BO card. The reflection coefficient method is used.

Parameters:

FLAGRF
- 0: No ground plane (or use one of the other ground plane options — such as the GF card for an exact model of real ground, or the SY card for a perfect ground plane). This option is used to switch off the reflection ground if the effect of different grounds are considered in a single input file.
- 1: Use the reflection coefficient ground plane approximation with the material parameters specified in the remaining input fields.
- 2: Use an ideal electric ground in the plane $z = 0$. In this case the remaining parameters are ignored.
- 3: Use an ideal magnetic ground in the plane $z = 0$. Also in this case the remaining parameters are ignored.

EPSRF — The relative dielectric constant ε_r of the ground.

SIGRF — The conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the ground.

MUERF — The relative permeability μ_r of the ground.

TANDMUERF — Magnetic loss factor $\tan\delta_\mu$ of the ground (the complex permeability is then given by $\mu = \mu_0\mu_r\left(1 - j\tan\delta_\mu\right)$).

TANDEPSRF — Electric loss factor $\tan\delta$ (an alternative way to specify the conductivity σ — the two loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$ and they have different frequency behaviour).

It should be noted that it is not possible to calculate the fields below the ground plane, i.e. it is not possible to calculate the fields in the region $z < 0$. In addition all structures must be in the region $z > 0$. If calculations in the ground are required, for example when there are structures below ground, the use of the exact Sommerfeld integrals (GF card) is recommended.

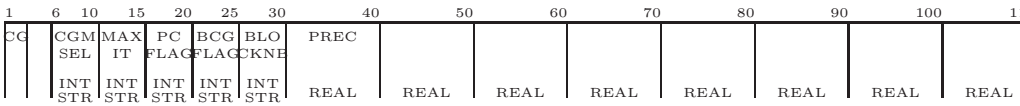
When using a perfect electric or magnetic reflection coeffiecient ground plane, structures can be arbitrarily close to the ground (while remaining above it). Note that structures cannot make electrical contact with the ground. If this is required electric symmetry should be used (see the SY card in section 8.2.34).

If real ground parameters are used the reflection coefficient approximation is more accurate for structures further from the ground plane. Typically structures should not be closer than about $\frac{\lambda}{10}$ (FEKO will give a warning if this is the case).

Note that for a perfect electrical ground (FLAGRF = 2) conducting structures, such as segments or triangles, cannot be electrically connected to the ground plane. If this is required, the ground plane should be realised with electrical symmetry (SY card).

A dielectric ground (real earth) can only be used with bodies treated with MoM, PO or the hybrid MoM/PO, i.e. the hybrid MoM/UTD method cannot be used in the presence of a real ground.

9.2.18 CG Card



Here the method used to solve the matrix equation may be chosen.

Parameters:

CGMSEL
- 0: Gauss Elimination, according to LINPACK, is used.
- 1: The CGM (*Conjugate Gradient Method*) is used.
- 3: The BCG (*Biconjugate Gradient Method*) is used.
- 4: Iterative algorithm with band matrix decomposition.
- 5: Gauss elimination, according to LAPACK, is used.
- 6: The block Gauss algorithm is used (in case the matrix has to be saved on the hard disk).
- 7: CG using PIM (parallel iterative methods).
- 8: Bi-CG using PIM.
- 9: CGS using PIM.
- 10: Bi-CGSTAB using PIM.
- 11: RBi-CGSTAB using PIM.
- 12: RGMRES using PIM.
- 13: RGMRESEV using PIM.
- 14: RGCR using PIM.
- 15: CGNR using PIM.
- 16: CGNE using PIM.
- 17: QMR using PIM.
- 18: TFQMR using PIM.
- 19: Parallel LU-decomposition with ScaLAPACK (solution in main memory) or with out-of-core ScaLAPACK (solution with the matrix stored on hard disk). This is the default option for parallel solutions and normally the user need not change it.
- 20: QMR using QMRPACK.

MAXIT The maximum number of iterations for the iterative techniques.

PCFLAG Determines the type of preconditioning:

- 0: No preconditioning is used.
- 1: Scaling the matrix $[A]$, so that the elements on the main diagonal are all normalised to one.
- 2: Scaling the matrix $[A]^H [A]$, so that the elements on the main diagonal are all normalised to one.
- 4: Block-Jacobi preconditioning with block size BLOCKNB for a PIM method. The inverses of the preconditioner are calculated and applied during every iteration step. For performance reasons PCFLAG=64 is recommended.
- 8: Block-Jacobi preconditioning of the matrix with block size BLOCKNB before beginning the iteration with a PIM method (i.e. before executing any matrix multiplication). For performance reasons PCFLAG=64 is recommended.
- 16: Symmetrisation of the matrix (in FEKO, a Galerkin technique is used with different integration algorithms, so that the resulting matrix is non-symmetrical).
- 32: Neumann polynomial preconditioning.
- 64: Block-Jacobi preconditioning, where for each block a LU-decomposition is computed in advance, and during the iterations a fast backward substitution is applied.
- 128: Incomplete LU-decomposition ILU(0) preconditioning.

BCGFLAG Parameters for the BCG (when CGMSEL=3):

- 1: Fletcher's method.
- 2: Jacobs' method.
- 3: Fletcher's method, pre-iteration using Fletcher's method.
- 4: Feltcher's method, pre-iteration using Jacobs' method.
- 5: Jabobs' method, pre-iteration using Fletcher's method.

BLOCKNB The block size to be used for LU-decomposition with LAPACK (CGMSEL=5) as well as for the Block-Jacobi preconditioning. When nothing is specified, appropriate standard values are used for LAPACK and the block preconditioners.

PREC Termination criterion for the normalised residue when using iterative methods.

Normally the CG card should not be used, in which case an optimum solution, based on a LU-decomposition, is selected (normally CGMSEL=5 for the sequential version of FEKO and CGMSEL=19 for the parallel version for the solution using main memory; when storing the data on disk a similar process is employed automatically). The CG card is only relevant in special circumstances, for example to select an iterative solution process.

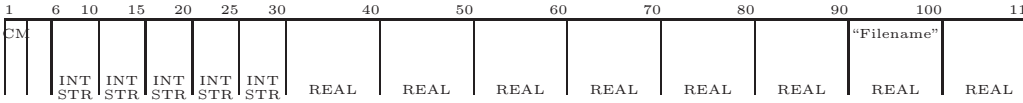
If more than one direction of incidence of a plane wave has been defined in the A0 card, the Gauss Elimination with CGMSEL=0 or 5 should be used, because an LU-decomposition of the matrix is stored. In such a case only a fast, backward substitution has to be done for each direction of incidence.

If CGMSEL=0 or 5 is selected, then LU-decomposition is done. The LU-decomposition is then stored instead of the original matrix. It is therefore not possible to use an iterative method to solve the matrix later without having to recalculate the elements.

Previous experience has shown that preconditioning always accelerates convergence. It is thus suggested that calculations be done with PCFLAG=1 or PCFLAG=2.

The iterative procedures using PIM (CGMSEL=7 to 18) only converge satisfactorily under special circumstances. The use of these techniques is not recommended, and it is therefore only available in Superuser Mode of the FEKO (see also SU card).

9.2.19 CM Card



This card is used to couple FEKO with the transmission line simmulation program CableMod[11] to calculate the coupling of electromagnetic fields into transmission lines. (The AC card is used for the case of radiation by these lines.)

The only parameter of this card is the file name of a `*.rsd` file created by CableMod (enclosed in double quotation marks and starting at or after column 91). The `*.rsd` file contains geometry of the line. With the CM card FEKO calculates the electric and magnetic near field at points along the line and write these to a `*.isd` file for further processing by CableMod. (The `*.isd` file also contains additional data required by CableMod, for example the frequencies that were used during the solution.)

The complete geometry (without the transmission line) as well as the frequency and excitation (Ax cards) must be defined in FEKO.

[11] To use the CableMod interface this module must be activated, if required please contact EMSS.

9.2.20 CO Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
CO		LAB	DO COVR	N			CDICKE	CMUER	CEPSR	CSIGMA	CRHO	CTAND MUE	CTAND EPS	
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card specifies a dielectric or magnetic coating of wire segments or triangle elements. The coating applies to all calculations following the CO card.

Parameters:

LAB — All segments or triangles with this label are coated.

DOCOVR — 0: No coating present (as if the relevant label has no CO card).
1: Coated wires using the Popovic formulation are used (see description below).
2: Coated wires using the volume equivalence theorem are applied (see description below).
3: Electrically thin multilayer dielectric/magnetic coating on surface triangle elements.
4: Multilayer dielectric/magnetic coating on surface triangle elements.

N — Number of layers. This field is only applicable for coatings on triangle elements, for wires only one layer is permitted.

CDICKE — Thickness of the outer layer in m (it is scaled by the SF card). For wires this is the radius of the coating less the radius ϱ of the wire-core.

CMUER — Relative permeability μ_r of the outer layer.

CEPSR — Relative permittivity ε_r of the outer layer.

CSIGMA — Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the outer layer.

CRHO — Wire radius ϱ of the metallic wire, without layers, in m (it is scaled by the SF card). This overrides the values specified with the IP card. This field is only applicable to wire coatings.

CTANDMUE — Magnetic loss factor $\tan\delta_\mu$ of the outer layer (the complex permeability is then given by $\mu = \mu_0\mu_r\left(1 - j\tan\delta_\mu\right)$).

CTANDEPS — Electric loss factor $\tan\delta$ of the outer layer (an alternative way to specify the conductivity σ, the two loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$ and have different frequency behaviour).

For DOCOVR=3 or 4 there follow N-1 more lines with only the parameters CDICKE, CMUER, CEPSR, CSIGMA, CTANDMUE and CTANDEPS of the remaining layers. Note that the layers are symmetric on both sides of the conductor and the last input line is closest to the conductor.

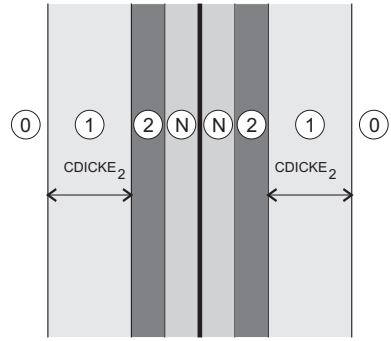


Figure 9-17: Double sided coating of surface elements. Note that the coating should be thin relative to the triangle dimentions. It is enlarged here for visualisation.

With DOCOVR, the user can chose between two different formulations for the treatment of the dielectric/magnetic coatings on wires or multilayer dielectric/magnetic coatings on surface triangles. These are:

- DOCOVR=1 (method using Popovic's formulation)
  In this case, the radius of the metallic core is changed internally to model the change in the capacitive loading of the wire and a corresponding inductive loading is added. The following restrictions apply:
  - The loss factor $\tan\delta$ of the layer (which is calculated from the conductivity σ and the relation $\tan\delta = \frac{\sigma}{\omega\epsilon_r\epsilon_0}$) has to be identical to the loss factor of the surrounding medium (specified with the EG card, usually free space)
  - Due to the change in the radius of the metallic core, no SK card should be active for the same label, otherwise the skin effect and/or the ohmic losses refers to the wire with changed radius.
  - For pure dielectric layers (i.e. the relative permeability μ_r of the layer equals that of the surrounding medium) the option DOCOVR=2 rather recommended.
- DOCOVR=2 (method using the volume equivalence theorem)
  Here the radius of the metallic wire is retained. The effect of the dielectric layer is accounted for by a volume polarisation current. The only restriction of this method is that the layer may not be magnetic in nature (i.e. the relative permeability μ_r of the coating must be the same as that of the surrounding medium).

- DOCOVR=3 (electrically thin coating on surface triangles)
  This option allows the user to add multilayer dielectric/magnetic coatings on surface elements with label LAB. The layers may have different permittivity and permeability, but the total coating must be electrically (i.e. relative to the wavelength in the coating) as well as geometrically (see the requirements below) thin.

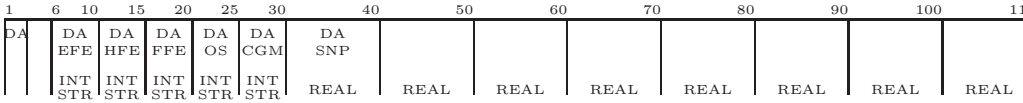
- DOCOVR=4 (dielectric/magnetic coating on surface triangles)
  This option allows electrically thick dielectric/magnetic coatings on surface elements with label LAB. Here it is only required that the total coating must be geometrically thin, i.e. it must be thin relative to the triangle size (and thus also to the free space wavelength) as well as the radius of curvature of the surface. This option may only be applied to elements treated with PO.

Note that for DOCOVR=1 or 2 no surface triangles with element LAB are allowed. Likewise with DOCOVR=3 or 4, no segments with label LAB are allowed.

If DOCOVR=4 is used, it must remain consistent for the whole FEKO run. Thus one may not have DOCOVR=4 for on some triangles for one solution and then add DOCOVR=4 to other triangles or remove it from some previously coated triangles for any subsequent run. It is, however, allowed to change the thickness and the medium parameters of the coating.

If a scaling factor (SF card) is present it affects both the thickness CDICKE and the wire radius CRHO.

9.2.21 DA Card



With this card some data like near fields or S-parameters can be exported to additional ASCII files. The card allows to switch this export on and off, and affects only cards for the computation of, for example, near fields or S-parameters that follow the DA card. By default, all export is off.

Parameters:

DAEFE	1: The electric field strength is stored in a `*.efe` file.
	0: The `*.efe` file is not stored.
DAHFE	1: The magnetic field strength is stored in a `*.hfe` file.
	0: The `*.hfe` file is not stored.
DAFFE	1: The far field is stored in a `*.ffe` file.
	0: The `*.ffe` file is not stored.
DAOS	1: The currents are stored in a `*.os` file.
	0: The `*.os` file is not stored.
DACGM	1: The residue from the iterative algorithm used to solve the matrix equation is stored in a `*.cgm` file.
	0: The `*.cgm` file is not stored.
DASNP	1: The S-parameters are written to a file in Touchstone `*.SnP` format. The `n` here gives the number of ports.
	0: The `*.SnP` file is not stored.

The exact description of the files can be found in section 2.2. More than one DA card is allowed in one input file. Thus, using the following control card sequence

```
DA      1
FE      1   ...
DA      0
FE      1   ...
```

the electric fields calculated with the first FE card are written to the `*.efe` file, but not those of the second FE card. The structure of the single files is described below:

***.efe file** When calculating the electric near field in Cartesian coordinates, there are columns with the position in x,y and z as well as the field components E_x, E_y and E_z in complex form. In cylindrical coordinate the columns consist of the following r, φ, z, E_r, E_φ and E_z. In spherical coordinates the columns consist of the following r, ϑ, φ, E_r, E_ϑ, E_φ.

***.hfe file** Same form as the `*.efe` file, except that the magnetic field values are given.

***.ffe file** Here the angles ϑ and φ are given. Following them are the field components E_ϑ and E_φ in complex form, with the factor $\frac{e^{-j\beta_0 R}}{R}$ left out as in the output file. Then the gain in dB, separated into the polarisation directions or the radar cross section, is given.

***.os file** First the triangle numbers and the centroids x,y and z are given as well as the complex current densities J_x, J_y and J_z at the centroid. Then there are three values which give the absolute value of the current density at the three corner points, as averaged over all triangles, that are adjacent to the corner points. The next three complex values are the components of the complex current density vector $\vec{J}$ for the first corner point of the triangle. The following groups of three are the values for the second and third corner points of the triangles.

After the current values in the triangles, the columns containing the data for the segments follow. For each segment a segment number is given, the centroid x,y and z as well as the current I_x, I_y and I_z flowing in the segment.

***.cgm file** In this file the number of iterations is given and the resulting residue from the iterative solving process of the matrix equation.

***.SnP file** The Touchstone S-parameter filename contains the number of ports in the model. The extension is `*.s1p` for a 1-port, `*.s2p` for a 2-port and so on. For 10-port and larger structures the `p` is dropped, for example `*.s12` for a 12-port. The file contains a header (following the `#` character) which specifies the frequency unit, the parameter type, the data format and the normalising impedance for all the ports. This is followed by the data lines (which may be repeated for multiple frequencies):

1-port:	f,	$\lvert S_{11}\rvert$, $\angle S_{11}$			
2-port:	f,	$\lvert S_{11}\rvert$, $\angle S_{11}$,	$\lvert S_{21}\rvert$, $\angle S_{21}$,	$\lvert S_{12}\rvert$, $\angle S_{12}$,	$\lvert S_{22}\rvert$, $\angle S_{22}$
3-port:	f,	$\lvert S_{11}\rvert$, $\angle S_{11}$,	$\lvert S_{12}\rvert$, $\angle S_{12}$,	$\lvert S_{13}\rvert$, $\angle S_{13}$	
		$\lvert S_{21}\rvert$, $\angle S_{21}$,	$\lvert S_{22}\rvert$, $\angle S_{22}$,	$\lvert S_{23}\rvert$, $\angle S_{23}$	
		$\lvert S_{31}\rvert$, $\angle S_{31}$	$\lvert S_{32}\rvert$, $\angle S_{32}$,	$\lvert S_{33}\rvert$, $\angle S_{33}$	
4-port:	f,	$\lvert S_{11}\rvert$, $\angle S_{11}$,	$\lvert S_{12}\rvert$, $\angle S_{12}$,	$\lvert S_{13}\rvert$, $\angle S_{13}$,	$\lvert S_{14}\rvert$, $\angle S_{14}$
		$\lvert S_{21}\rvert$, $\angle S_{21}$,	$\lvert S_{22}\rvert$, $\angle S_{22}$,	$\lvert S_{23}\rvert$, $\angle S_{23}$,	$\lvert S_{24}\rvert$, $\angle S_{24}$
		$\lvert S_{31}\rvert$, $\angle S_{31}$,	$\lvert S_{32}\rvert$, $\angle S_{32}$,	$\lvert S_{33}\rvert$, $\angle S_{33}$,	$\lvert S_{34}\rvert$, $\angle S_{34}$
		$\lvert S_{41}\rvert$, $\angle S_{41}$,	$\lvert S_{42}\rvert$, $\angle S_{42}$,	$\lvert S_{43}\rvert$, $\angle S_{43}$,	$\lvert S_{44}\rvert$, $\angle S_{44}$

where $|S_{11}|$ is the absolute value and $\angle S_{11}$ the phase (in degrees) of the given parameter. Note that the 2-port file is formatted on a single line and in a different order than the rest.

For all files except `*.SnP` the data is in rows, i.e. each new set of data is in a new row. Complex numbers are given in the normal FORTRAN format `(Real,Imaginary)` output, e.g. `(-6.956E-03,1.034539E-07)`.

9.2.22 DI Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
DI		MED					EPSR	MUER	SIGMA	TAND MUE	TAND EPS	R_6		
		INT	INT	INT	INT	INT	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL
		STR	STR	STR	STR	STR								

Here the electric characteristics of the dielectric and magnetic bodies are entered, when using the surface current method.

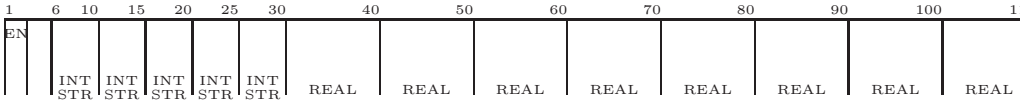
Parameters:

MED	This gives the index of the medium as used at the ME card. Zero indicates the surrounding free space medium, and any other value the respective medium.
EPSR	Relative dielectric constant ε_r of the medium MED.
MUER	Relative permeability μ_r of the medium MED.
SIGMA	Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the medium MED.
TANDMUE	Magnetic loss factor $\tan\delta_\mu$ of the medium MED (the complex permeability is then given by $\mu = \mu_0\mu_r\,(1 - j\tan\delta_\mu)$).
TANDEPS	Electric loss factor $\tan\delta$ of the medium MED (this is alternative way to specify the conductivity σ — the two loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$ and have different frequency behaviour).
R_6	The medium density (in kg/m^3). This parameter is not used in FEKO, but is written to the output file and used for SAR calculation during post processing.

Note that for backwards compatibility to older FEKO versions (where only one medium was allowed and the parameter MED did not exist), if the parameter MED is not specified (i.e. input field is empty), then this defaults to MED=1 and hence sets the material parameters of medium 1.

Using MED=0 will overwrite the default free space parameters that might have been set at the EG card or at a previous GF card for the free space Green's function.

9.2.23 EN Card



This card indicates the end of the input file. It is essential and has no parameters.

9.2.24 FE Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
FE		FEL TYP	ANZX	ANZY	ANZZ	FEL KOR	X0	Y0	Z0	DX	DY	DZ		R8
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card controls the calculation of the near fields.

Parameters:

FELTYP
- 0: Field is not calculated.
- ±1: Calculate the electric field in free space.
- ±2: Calculate the magnetic field in free space.
- ±3: Calculate both electric and magnetic fields in free space.
- 7: Outputs the electric field as well as the SAR-values in the dielectric volume elements. The other parameters (e.g. observation points) are not required.
- ±10: Compute the magnetic vector potential $\vec{A}$.
- ±11: Compute the gradient of the scalar electric potential $\nabla\varphi$.
- ±12: Compute the electric vector potential $\vec{F}$.
- ±13: Compute the gradient of the scalar magnetic potential $\nabla\psi$.

Negative sign: only the scattered part of the field/potential (no source contribution) is written to the output file.

Positive sign: The total field/potential, the sum of the scattered and source contibutions, is written to the output file.

Values for |FELTYP| = 4,5 and 6 are obsolete and should be replaced with 1,2 and 3 respectively.

ANZX Number of field points in x or r direction (see FELKOR below). For FELKOR=6 ANZX and ANZY are not used.

ANZY Number of field points in y, φ or ϑ direction.

ANZZ Number of field points in z, φ, x or y direction. For FELKOR=6 this parameter specifies the total number of points).

FELKOR indicates the coordinate system used, in which the near field is to be calculated:
- 0: Cartesian coordinates (x, y, z).
- 1: Cylindrical coordinates (r, φ, z).
- 2: Spherical coordinates (r, ϑ, φ).
- 3: Cylindrical coordinates (r, φ, x) around the x axis.
- 4: Cylindrical coordinates (r, φ, y) around the y axis.
- 5: Conical coordinates (φ, z) around the z axis.
- 6: Irregular points in Cartesian coordinates (x, y, z).

X0	x or r coordinate (dependent on FELKOR) of the first observation point.
Y0	y or φ or ϑ coordinate of the first observation point.
Z0	z or φ coordinate of the first observation point.
DX	Increment size in x or r direction (dependent on FELKOR).
DY	Increment size in y or φ or ϑ direction.
DZ	Increment size in z or φ direction.
R8	If R8 is set to -1, the old format of the near field is used in the `*.out` output file. This should only be used for compatibility with third party post processors. Note that WinFEKO and Graph-FEKO cannot extract SAR values from near fields in this format.

Note that all coordinates are metre and all angles in degrees. Scaling with the SF card is only applicable when the option SKALFLAG=1 is selected (highly recommended) — in this case coordinates must be in metre after scaling.

Potentials cannot be computed with the FE card if UTD or PO is used. Also only the free space Green's function is supported, but not the Green's functions for layered spheres or multilayered planar media.

If the total potentials are requested, the potentials for the sources are added. These are not available for a plane wave (A0 card) or an impressed radiation pattern (AR card) and FEKO will give an error. For a magnetic dipole (A6 card) if the electric ring current model is used one will get $\vec{A}$, and if the magnetic current model is used one may compute $\vec{F}$ and $\nabla\psi$, all the other potentials are zero.

If one requests `*.efe` and/or `*.hfe` files with the DA-card, then $\vec{A}$ and $\nabla\varphi$ are written to the `*.efe` file, while $\vec{F}$ and $\nabla\psi$ are written to the `*.hfe` file.

The different possibilities of FELKOR are described on the following pages.

- FELKOR=0 (Cartesian coordinates x, y, z)

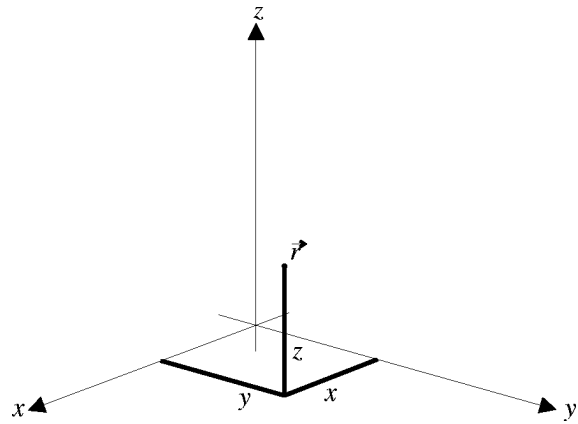


Figure 9-18: Field calculation in the near field (FELKOR=0)

Observation Point:

$$\vec{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Unit vectors of the coordinate system:

$$\hat{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \hat{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \hat{z} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

- FELKOR=1 (Cylindrical coordinates around z axis r, φ, z)

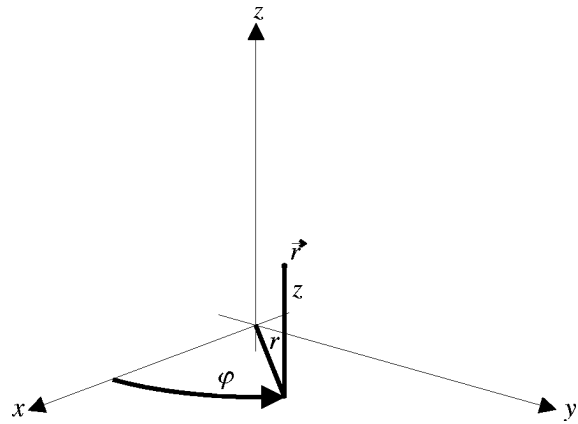


Figure 9-19: Field calculation in the near field (FELKOR=1)

Observation Point:

$$\vec{r} = \begin{bmatrix} r\,\cos\varphi \\ r\,\sin\varphi \\ z \end{bmatrix}$$

Unit vectors of the coordinate system:

$$\hat{r} = \begin{bmatrix} \cos\varphi \\ \sin\varphi \\ 0 \end{bmatrix} \qquad \hat{\varphi} = \begin{bmatrix} -\sin\varphi \\ \cos\varphi \\ 0 \end{bmatrix} \qquad \hat{z} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

- FELKOR=2 (Spherical coordinates r, ϑ, φ)

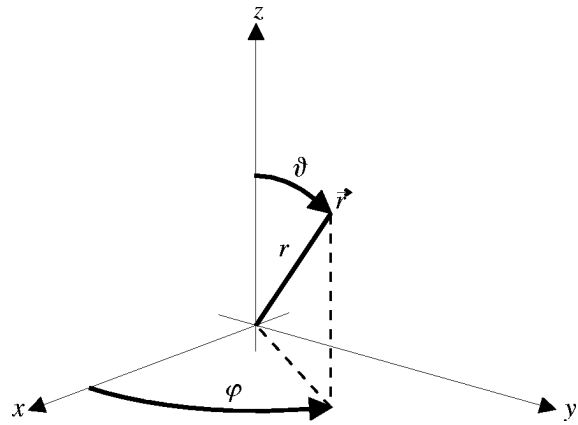


Figure 9-20: Field calculation in the near field (FELKOR=2)

Observation Point:

$$\vec{r} = \begin{bmatrix} r\,\sin\vartheta\cos\varphi \\ r\,\sin\vartheta\sin\varphi \\ r\,\cos\vartheta \end{bmatrix}$$

Unit vectors of the coordinate system:

$$\hat{r} = \begin{bmatrix} \sin\vartheta\cos\varphi \\ \sin\vartheta\sin\varphi \\ \cos\vartheta \end{bmatrix} \qquad \hat{\vartheta} = \begin{bmatrix} \cos\vartheta\cos\varphi \\ \cos\vartheta\sin\varphi \\ -\sin\vartheta \end{bmatrix} \qquad \hat{\varphi} = \begin{bmatrix} -\sin\varphi \\ \cos\varphi \\ 0 \end{bmatrix}$$

- FELKOR=3 (Cylindrical coordinates around the x axis r, φ, x)

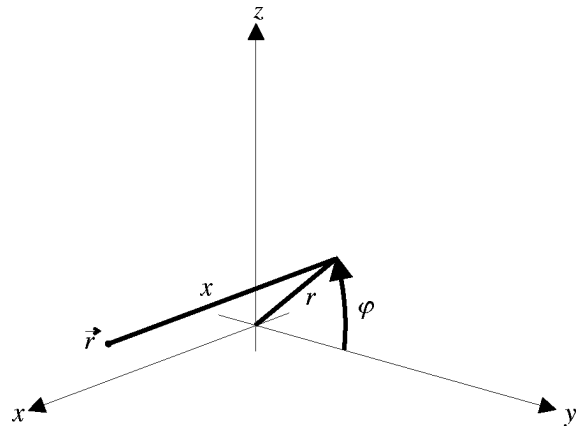


Figure 9-21: Field calculation in the near field (FELKOR=3)

Observation Point:

$$\vec{r} = \begin{bmatrix} x \\ r\,\cos\varphi \\ r\,\sin\varphi \end{bmatrix}$$

Unit vectors of the coordinate system:

$$\hat{r} = \begin{bmatrix} 0 \\ \cos\varphi \\ \sin\varphi \end{bmatrix} \qquad \hat{\varphi} = \begin{bmatrix} 0 \\ -\sin\varphi \\ \cos\varphi \end{bmatrix} \qquad \hat{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

- FELKOR=4 (Cylindrical coordinates around the y axis r, φ, y)

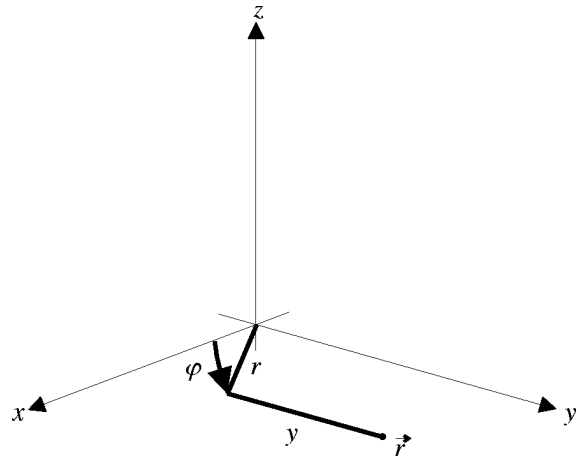


Figure 9-22: Field calculation in the near field (FELKOR=4)

Observation Point:

$$\vec{r} = \begin{bmatrix} r\,\cos\varphi \\ y \\ -r\,\sin\varphi \end{bmatrix}$$

Unit vectors of the coordinate system:

$$\hat{r} = \begin{bmatrix} \cos\varphi \\ 0 \\ -\sin\varphi \end{bmatrix} \qquad \hat{\varphi} = \begin{bmatrix} -\sin\varphi \\ 0 \\ -\cos\varphi \end{bmatrix} \qquad \hat{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

- FELKOR=5 (Conical coordinates around the z axis φ, z)

  This option is similar to the field calculation in cylindrical coordinates around the z axis, where the radius r changes with the height z

$$r(z) = r_0 + \frac{\Delta r}{\Delta z} \cdot (z - z_0),$$

  and z lies within the range $z_0 \ldots z_0 + n_z \cdot \Delta z$. The parameters are defined in the following fields:

r_0	X0
φ_0	Y0
z_0	Z0

Δr	DX
$\Delta\varphi$	DY
Δz	DZ
n_r	ANZX
n_φ	ANZY
n_z	ANZZ

It should be noted, that $n_r = ANZX$ must be 1.

Observation Point:

$$\vec{r} = \begin{bmatrix} \left(r_0 + \frac{\Delta r}{\Delta z} \cdot (z - z_0)\right) \cdot \cos\varphi \\ \left(r_0 + \frac{\Delta r}{\Delta z} \cdot (z - z_0)\right) \cdot \sin\varphi \\ z \end{bmatrix}$$

Unit vectors of the coordinate system:

$$\hat{r} = \begin{bmatrix} \cos\varphi \\ \sin\varphi \\ 0 \end{bmatrix} \qquad \hat{\varphi} = \begin{bmatrix} -\sin\varphi \\ \cos\varphi \\ 0 \end{bmatrix} \qquad \hat{z} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

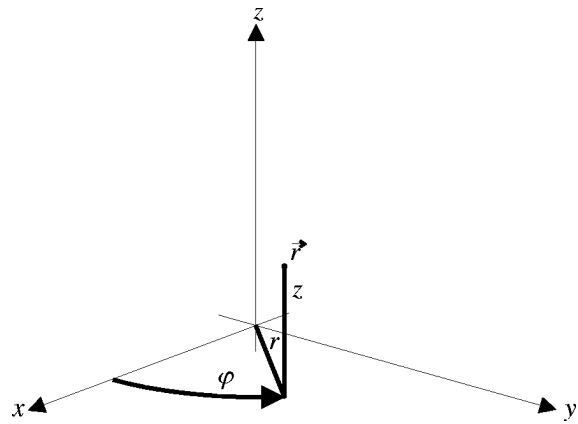


Figure 9-23: Field calculation in the near field (FELKOR=5)

- FELKOR=6 (irregular points in Cartesian coordinates)

  This option allows computation of the near fields at arbitrary points (defined in Cartesian coordinates). If this is the case only the following parameters of the FE card are of any significance:

FELTYP	as described above
ANZZ	the total number of field points
FELKOR	must be set to 6

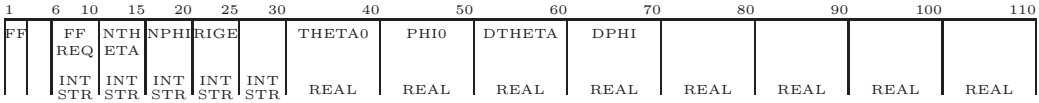
  The FE card is followed by exactly ANZZ lines. Each of these specify the (x, y, z) coordinates of one point in the first three real parameter fields. A simple example with three points are as follows:

```
FE   1              3    6
                              1.24       -0.2       0.7
                              3.87       -0.25      0.8
                              2.5        0.2        1.2
```

If a ground plane is used, a calculation of the near fields in the ground plane is not possible. The observation points in the area $z < 0$ are not taken into account.

It should be noted that the coordinates may have an offset (OF card). Thus the near field on the surface of a sphere can be calculated, with the centre of the sphere not being located at the origin of the coordinate system.

9.2.25 FF Card



This card controls the calculation of the far fields in spherical coordinates.

Parameters:

FFREQ
- 0: No calculation is done.
- ±1: The far field is calculated using the following parameters.
- ±2: The far field is calculated only in the incident direction (used, for example, to calculate monostatic RCS).
- ±3: The far field is calculated as for FFREQ=±1, but it is not written to the output file in order to limit the size thereof. This option is meaningful when the individual values of the field strength (such as directivity and gain) is not required, but the total radiated power has to calculated from the integral of the Poynting vector (see the discussion below). If a `*.ffe` file has been requested with the DA card, it is written as if for FFREQ=±1. (Unless it is turned off by a later DA card.)

Positive values of FFREQ: The total field is calculated. This includes all source contributions except plane wave excitations.

Negative values of FFREQ: The field radiated by the impressed sources (such as Hertzian dipoles) are not included. This option is only meaningful if only the scattered field is required. Normally one would use positive values of FFREQ.

NTHETA The number of observation points in the ϑ direction. An empty field i.e. NTHETA=0 will be set to NTHETA=1.

NPHI The number of observation points in the φ direction. An empty field i.e. NPHI=0 will be set to NPHI=1.

RIGE
- 0: The directivity of an antenna is calculated.
- 1: The gain of an antenna is calculated.

THETA0 ϑ coordinate ϑ_0 in degrees of the first observation point.

PHI0 φ coordinate φ_0 in degrees of the first observation point.

DTHETA Increment $\Delta\vartheta$ in degrees of the angle ϑ.

DPHI Increment $\Delta\varphi$ in degrees of the angle φ.

When calculating the monostatic radar cross section for a number of directions of incidence, the parameter FFREQ=2 is necessary, otherwise FFREQ=1 can be used.

When using the FF card with NTHETA ≥ 2 and NPHI ≥ 2, then the Poynting vector is integrated over the two spherical segments

- $\vartheta_0 - \frac{1}{2} \cdot \Delta\vartheta \leq \vartheta \leq \vartheta_0 + (\texttt{NTHETA} - \frac{1}{2}) \cdot \Delta\vartheta$ and $\varphi_0 - \frac{1}{2} \cdot \Delta\varphi \leq \varphi \leq \varphi_0 + (\texttt{NPHI} - \frac{1}{2}) \cdot \Delta\vartheta$
- $\vartheta_0 \leq \vartheta \leq \vartheta_0 + (\texttt{NTHETA} - 1) \cdot \Delta\vartheta$ and $\varphi_0 \leq \varphi \leq \varphi_0 + (\texttt{NPHI} - 1) \cdot \Delta\varphi$

If only the integrated power is of concern, the option FFREQ=$\pm$3 may be used to reduce the size of the output file.

In the case of an antenna the power provided by the voltage sources must be equal to the radiated power over the whole sphere. The total radiated power can be calculated using for instance the following commands:

```
** Far field integration in angular increments of #delta (in degrees)
#delta = 5
#nt = 180 / #delta + 1
#np = 360 / #delta + 1
FF   3    #nt  #np  0         0         0         #delta    #delta
```

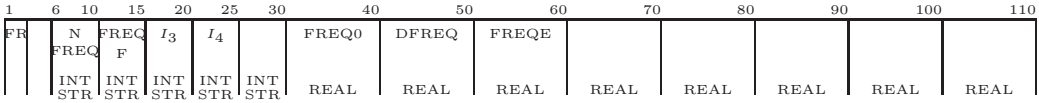
The output in the `*.out` file then reads for example

```
Integration of the normal component of the Poynting vector in the angular
grid DTHETA =    5.00 deg. and DPHI =    5.00 deg. (    2701 sample points)
  angular range THETA        angular range PHI       radiated power
 -2.50 .. 182.50 deg.      -2.50 .. 362.50 deg.     5.60499E-03 Watt
  0.00 .. 180.00 deg.       0.00 .. 360.00 deg.     5.52821E-03 Watt
```

If the problem is symmetrical, it is not necessary to carry out the integration over the complete sphere. If there are three planes of symmetry (as for a simple dipole in free space) the integration only needs to be done over an eighth of the sphere. The power then has to multiplied by 8.

If a ground plane has been specified, the calculation of the far fields below the ground plane is not possible. Observation points with $z < 0$ will thus be ignored.

9.2.26 FR Card



This card sets the frequency (in Hz), at which the solution will be obtained. This may be a single frequency or a loop of discrete frequencies (linear or multiplicative stepping). One can also obtain a continuous solution in a given frequency band with adaptive frequency interpolation. If this is used, only one FR card is allowed.

Parameters:

FREQF
- 0: Loop using discrete frequency points. Consecutive frequencies differ with DFREQ, i.e. the new frequency value will be the value of DFREQ added to the previous value.
- 1: Loop using discrete frequency points. The frequencies differ by the factor DFREQ, i.e. the new frequency is DFREQ times the previous value.
- 2: Use an adaptive frequency interpolation technique to obtain a continuous representation of the results in the given frequency band.

for FREQF=0 or 1:

NFREQ Number of frequencies to be examined.

FREQ0 Starting frequency in Hz.

DFREQ The frequency increment in Hz or the multiplication factor (when FREQF=1).

FREQE Optional ending frequency in Hz. If this is specified, DREQF must be empty – it will be calculated as shown below.

for FREQF=2:

NFREQ Maximum number of discrete frequency points in this frequency band at which FEKO may be executed (limitation to avoid convergence problems). If left empty, the default value of 1000 will be used.

I_3 This field is only relevant when the CM card is used to create a `*.isd` file. The results are written to the `*.isd` file at I_3 discrete frequencies.

I_4 This field is only relevant when the CM card is used to create a `*.isd` file. It then specifies the type of frequency stepping of the discrete frequencies written to the `*.isd` file:
- 0: The discrete frequencies differ by a constant increment.
- 1: Successive frequencies are related by a constant factor.

FREQ0 Starting frequency in Hz.

DFREQ In order to obtain a continuous frequency response, the adaptive frequency interpolation technique obtains the solution at a set of discrete frequency points. They are automatically placed, for example using large frequency increments in regions with a smooth behaviour of the results, and much finer frequency increments close to resonances. Sometimes, for example when using a frequency dependent mesh, it can happen that the FEKO results versus frequency are not exactly continuous. To avoid that the adaptive algorithms gets stuck at such small discontinuities and tries to refine more and more (it will stop when NFREQ points are reached of course), one can set DFREQ to be the minimum allowable separation distance between neighbouring frequency sample points. The value of DFREQ must be smaller than the resolution required to solve, for example, sharp resonances. If left empty, the default is

1.0e-4*(FREQE-FREQ0).

FREQE Ending frequency in Hz (not optional in this case).

If a discrete loop with more that one frequency is required (FREQF≠2, NFREQ≠1) then either DFREQ or FREQE must specified, but not both. If the end frequency FREQE is specified, DFREQ is calculated from:

- for FREQF=0 (additive increments):

$$\text{DFREQ} = \frac{\text{FREQE} - \text{FREQ0}}{\text{NFREQ} - 1}$$

- for FREQF=1 (multiplicative increments):

$$\text{DFREQ} = \left(\frac{\text{FREQE}}{\text{FREQ0}}\right)^{\frac{1}{\text{NFREQ}-1}}$$

When writing results at discrete frequencies to a `*.isd` file, the frequency increment when $I_4 = 0$ is calculated similar to the case for FREQF= 0 above (with NFREQ replaced by I_3) and the factor for $I_4 = 1$ is similar to that for FREQF= 1.

If more than one frequency is to be examined, then all the control cards up to the next FR card or EN card will be read into a buffer and are executed for each frequency. More information can be found in section 9.1.

9.2.27 GF Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
GF		GF FLAG	I_2			I_5	R_1	R_2	R_3	R_4	R_5	R_6	R_7	
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With this card the Green's function may be selected. Currently the following Green's functions are supported:

- **Homogeneous Medium (GFFLAG=0)**

  This is the standard "free space" Green's function similar to when the GF card is not used. The medium is normally free space, but different parameters can be set with the EG card. The parameters as set by the EG card is retained only when all the real parameters R_1 to R_5 of the GF card is empty or zero. If any of the medium parameters below is set, *all* the EG card values are overridden. (Those parameters that are not specified will then default to the values given here.)

  Parameters:

  GFFLAG 0: Green's function for free space is used.

  R_1 Relative permittivity ε_r of the homogeneous medium (if this field is empty, $\varepsilon_r = 1$ is used).

  R_2 Relative permeability μ_r of the homogeneous medium (if this field is empty, $\mu_r = 1$ is used).

  R_3 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the homogeneous medium.

  R_4 Magnetic loss factor $\tan\delta_\mu$ of the homogeneous medium (the complex permeability is then given by $\mu = \mu_0\mu_r\,(1 - j\tan\delta_\mu)$; $\tan\delta_\mu$ is set to zero if this field is empty).

  R_5 Electric loss factor $\tan\delta$ of the homogeneous medium. This is an alternative way to specify the conductivity σ — the two loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$ but have different frequency behaviour. (If both R_3 and R_5 are empty or zero, $\tan\delta$ is set to zero).

- **Layered dielectric sphere (GFFLAG=1,2,4,5,6)**

  With this option it is, for example, possible to analyse a cellphone in front of a shell model of the human head very efficiently.

  Parameters:

  GFFLAG
  - 1: Green's function for a homogeneous dielectric sphere at the origin of the coordinate system is used.
  - 2: Green's function for a homogeneous dielectric sphere, that has been coated with a dielectric and is situated at the origin of the coordinate system.
  - 4: Green's functions for a homogeneous dielectric sphere, which consists of a core and three dielectric layers, at the origin of the coordinate system.
  - 5: As when GFFLAG=1 a Green's function for a homogeneous dielectric sphere at the coordinate origin is used, but in contrast to GFFLAG=1, metallic structures can be present in the inner parts of the sphere.
  - 6: Green's function for homogeneous dielectric sphere at the origin of the coordinates, which consists of a core and 2 layers. Metallic structures are allowed inside the sphere.

  I_5
  - 0: Use of interpolation (`*.gfe` and `*.gfh` files) to accelerate the computations.
  - 1: No interpolation used.

  R_5 Convergence criteria for the summation of the rows of Green's functions. If R_5 is 0 or undefined, a sensible standard criterion is used.

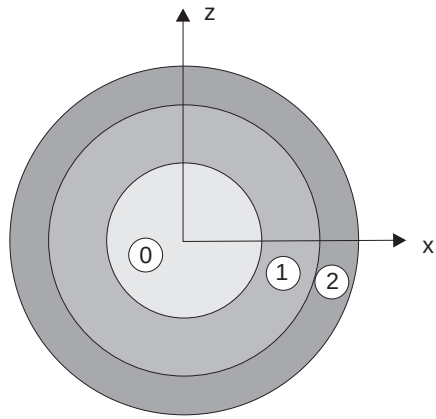


Figure 9-24: Example of a sphere with 3 layers (GFFLAG=6).

for GFFLAG=1:

- R_1 Sphere's radius in m (is scaled by the SF card).
- R_2 Relative dielectric constant ε_r.
- R_3 Relative permeability μ_r.
- R_4 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$.
- R_6 Electric loss factor $\tan\delta$ (an alternative to the conductivity σ — the two loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$).
- R_7 Magnetic loss factor $\tan\delta_\mu$ (the complex permeability is then given by $\mu = \mu_0\mu_r\left(1 - j\tan\delta_\mu\right)$).

for GFFLAG=2:

- R_1 Sphere's radius — core including the coating — in m (is scaled by the SF card).
- R_2 Relative dielectric constant ε_r of the coating.
- R_3 Relative permeability μ_r of the coating.
- R_4 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the coating.
- R_6 Electric loss factor $\tan\delta$ of the coating (an alternative to the conductivity σ — the loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$).
- R_7 Magnetic loss factor $\tan\delta_\mu$ of the coating (the complex permeability is then given by $\mu = \mu_0\mu_r\left(1 - j\tan\delta_\mu\right)$).

for GFFLAG=2 another line is specified as follows:

- R_1 Radius of the core in m (is scaled by the SF card).
- R_2 Relative dielectric constant ε_r of the core.
- R_3 Relative permeability μ_r of the core.
- R_4 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the core.
- R_6 Electric loss factor $\tan\delta$ of the core.
- R_7 Magnetic loss factor $\tan\delta_\mu$ of the core.

for GFFLAG=4:

- R_1 Sphere's radius — including the outermost layer — in m (is scaled by the SF card).
- R_2 Relative dielectric constant ε_r of the third (outer) layer.
- R_3 Relative permeability μ_r of the third layer.
- R_4 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the third layer.
- R_6 Electric loss factor $\tan\delta$ of the third layer (an alternative to the conductivity σ — the loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$).
- R_7 Magnetic loss factor $\tan\delta_\mu$ of the third layer (the complex permeability is then given by $\mu = \mu_0\mu_r\left(1 - j\tan\delta_\mu\right)$).

For GFFLAG=4 the GF card must be followed by three additional lines with the characteristics of the second and first layers as well as the core in this order.

for GFFLAG=5:

- R_1 Sphere's radius in m (is scaled by the SF card).
- R_2 Relative dielectric constant ε_r.
- R_3 Relative permeability μ_r.
- R_4 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$.
- R_6 Electric loss factor $\tan\delta$ (an alternative to specifying the conductivity σ — the two loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$).
- R_7 Magnetic loss factor $\tan\delta_\mu$ (the complex permeability is then given by $\mu = \mu_0\mu_r\left(1 - j\tan\delta_\mu\right)$).

for GFFLAG=6:

- R_1 Radius of the outer most layer in m (is scaled by the SF card).
- R_2 Relative dielectric constant ε_r of the second layer.
- R_3 Relative permeability μ_r of the third layer.
- R_4 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$of the third layer.
- R_6 Electric loss factor $\tan\delta$ of the third layer (an alternative to specifying the conductivity σ — the two loss terms are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$).
- R_7 Magnetic loss factor $\tan\delta_\mu$ of the third layer (the complex permeability is then given by $\mu = \mu_0\mu_r\left(1 - j\tan\delta_\mu\right)$).

For GFFLAG=6 another 2 lines (analogous to this one) have to be defined for the first layer (in the middle) and for the core.

The scaling factor that is entered by the SF card is applied to the radius. The parameters of the medium outside the sphere (usually free space) can be set with the EG card.

The Green's function for a homogeneous or layered dielectric sphere can be used with metallic structures (treated with the MoM) either inside or outside the sphere (but not for example a wire from inside to outside). It can be used with dielectric bodies treated with the volume equivalence principle (e.g. the hand of a user around a mobile phone), but the dielectric bodies must be outside the sphere.

An example of the use of the GF card for a sphere is given in `example_15` (*Examples Guide*).

- **Planar Multilayer Substrate (GFFLAG=10,11,12,13)**

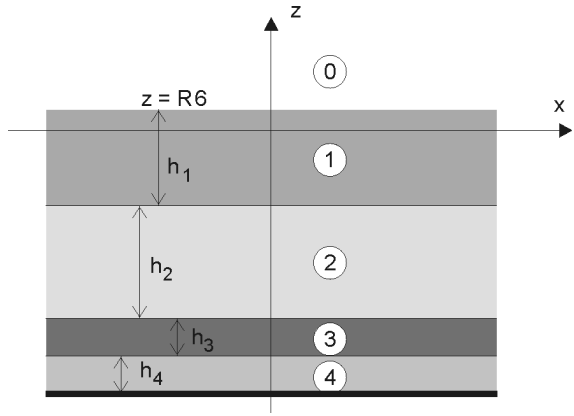


Figure 9-25: Example of a 4 layer substrate with a metallic ground plane.

Parameters:

GFFLAG
- 10: Green's function for planar layered media with a metallic ground plane at the bottom of layer I_2 (the top layer 0 extends to $z = +\infty$). Metallic structures or field computations are only supported above the ground plane. The ground plane is assumed to be infinitely large, a finite ground plane can be modelled directly in FEKO using triangular patches.
- 11: Layered dielectric media without metallic ground planes (the top layer 0 extends to $z = +\infty$, the last layer I_2 extends to $z = -\infty$).
- 12: Layered dielectric media with two metallic ground planes, the top one is located between layers 0 and 1, and the bottom one at the bottom of layer I_2. Metallic structures or field computations are only supported between the ground planes, not above the top one or below the bottom one (they are assumed to be infinitely large). The material parameters of layer 0 are irrelevant.
- 13: Layered dielectric media with a metallic ground plane at the top between layers 0 and 1. The last layer I_2 extends to $z = -\infty$. Metallic structures or field computations are only supported below the ground plane.

I_2 Number of layers in the substrate, in the example in figure 9-25 there are 4 layers. (Medium 0, or the half-space $z > R_6$ is not included in the number).

R_2 Relative permittivity ε_r of medium 0 (the half-space $z > R_6$).

R_3 Relative permeability μ_r of medium 0 (the half-space $z > R_6$).

R_4 Conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of medium 0 (the half-space $z > R_6$; see the comment below).

R_5 Electric loss factor $\tan\delta$ in medium 0 (the half-space $z > R_6$; see the comment below).

R_6 z coordinate of the upper boundary of layer 1 (the uppermost layer, see figure 9-25) in m (note that it is scaled by the SF card).

R_7 Magnetic loss factor $\tan\delta_\mu$ in medium 0 (the half-space $z > R_6$; the complex permeability is given by $\mu = \mu_0\mu_r\,(1 - j\tan\delta_\mu)$).

This line is followed by a further I_2 lines (i.e. I_2+1 lines in total) to specify the parameters of all the layers. The parameters are taken from top to bottom and are as follows:

R_1 The thickness of the layer in m (see also the parameter h_i in figure 9-25; note that it is scaled by the SF card). When the parameter GFFLAG=11 or 13 the last (bottom) layer is infinitely thick. The field R_1 can then be set to any value.

R_2 Relative permittivity ε_r of the layer.

R_3 Relative permeability μ_r of the layer.

R_4 Conductivity σ of the layer.

R_5 Electric loss factor $\tan\delta$ of the layer (see the comment below).

R_7 Magnetic loss factor $\tan\delta_\mu$ of the layer.

Losses in any layer, or the half-space $z > R_6$, may be specified through assignment of either the conductivity σ (parameter R_4) or by the electric loss factor $\tan\delta$ (parameter R_5). Only one of these input values may be used; the other must remain empty. They are related by $\tan\delta = \frac{\sigma}{\omega\varepsilon_r\varepsilon_0}$. It should be noted that, inside a frequency loop with varying ω, the first option is constant while second one is not.

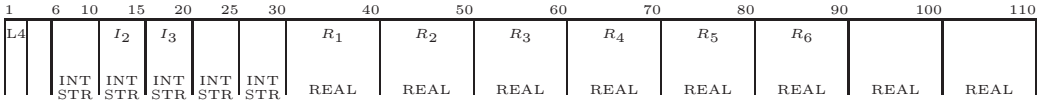
Metallic structures can have an arbitrary orientation (horizontal, vertical, and also diagonal). They can lie at an arbitrary position in one or more layers and can lie directly on the border of two layers (e.g. on the surface of a substrate). The only restriction is that no metallic segment or triangle may cross a boundary between layers, i.e. it must lie completely within one layer or at the boundary between layers.

If, for example, a metallic wire penetrates a multilayer substrate, the segmentation must be such that there is a node on each interface between layers. (See `example_31` in the *Examples Guide*.)

The following is not possible with this Green's function:

- dielectric ground
- hybrid MoM/PO method
- hybrid MoM/UTD method
- dielectric bodies with surface equivalence principle

9.2.28 L4 Card



This card can be used to add a load between a metallic triangle and the ground plane for the planar multilayer Green's function without having the requirement to model a vertical current element (analogous to the A4 excitation card).

Parameters:

- I_2 The label of the triangle to load. If there are more than one triangle with this label, the one with the highest element number is loaded. Alternatively, the user may set I_2=-1, and specify the Cartesian coordinates $x = R_3$, $y = R_4$ and $z = R_5$ of the load point. The triangle with the centre point closest to this point is loaded.
- I_3 0: The specified impedance refers directly to the load point (centroid of the triangle).
  1: The specified impedance refers to the metallic ground plane, and a transformation must be done to get the correct load impedance at the triangle.
- R_1 Real part of the load impedance (in Ω).
- R_2 Imaginary part of the load impedance (in Ω).
- R_3 Only if I_2=-1: the x coordinate of the load position in m.
- R_4 Only if I_2=-1: the y coordinate of the load position in m.
- R_5 Only if I_2=-1: the z coordinate of the load position in m.
- R_6 Only if I_3=-1: the radius of the load pin in m.
  (The values R_3, R_4, R_5 and R_6 is scaled by the SF card if SKALFLAG=1.)

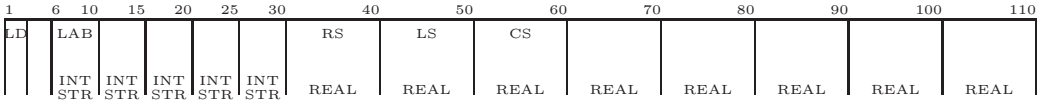
The implementation is such that if an L4 card is processed any existing load on that triangle is replaced. For example, the card sequence

```
L4    5      50        0
L4    5      20        0
```

will add a 20 Ω load to the triangle with label 5, and not a 70 Ω load.

It must also be noted that if the L4 card is used in conjunction with an A4 card (impressed current source), the load impedance of the L4 card is placed parallel to the input impedance of the A4 source (it has no effect if it is in series with the current source), i.e. the resulting input admittance is the sum of input admittance without the L4 load, and the admittance added by the L4 card.

9.2.29 LD Card



With this card it is possible to specify a distributed resistive, capacitive or inductive loading or even a series combination of these values for a segment.

Parameters:

LAB	All segments with this label are subjected to distributed loading.
RS	The distributed resistance in $\frac{\Omega}{\mathrm{m}}$.
LS	The distributed inductance in $\frac{\mathrm{H}}{\mathrm{m}}$.
CS	The distributed capacitance in $\frac{\mathrm{F}}{\mathrm{m}}$.

The combined impedance of the segment with length l is then

$$Z_s = \left(R' + j\omega L' + \frac{1}{j\omega C'}\right) \cdot l.$$

It should be noted that CS=0 is treated as infinite capacitance, such that the C' term may be neglected.

The LD card may be combined with the LP, LS, LZ and the SK cards, but only one LD card may be used per label. If a second LD card is used, it replaces the values entered by the first one. This card has no significance for surface elements, even when these are assigned the same label.

9.2.30 LE Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
LE		I_1	I_2	I_3			R	X						
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

With the LE card an edge between surface triangles can be loaded with an impedance $Z = R + jX$ (see the AE card for the excitation of such an edge) as shown in figure 9-26. As shown in the figure the edge can consist of several single edges — each of which should be common to triangles that have one of only two labels. One of these labels must be unique, i.e. only one triangle at each single edge should have this label. (See the AE card — section 9.2.12 — for a discussion of the allowed configurations.) Alternatively the edge can be along a connection between triangles and a polygonal UTD plate or a PEC ground plane, or it can be a microstrip feed line port. The impedance Z applies to the complete edge (i.e. all the single edges in parallel). The LE card can be combined with the AE card to specify both an impedance and a voltage source over the edge.

Parameters:

I_1/I_2 The meaning of these parameters depends on I_3.

I_3
- 0: Load the edge between the regions with labels I_1 and I_2 with a complex impedance.
- 2: Load the edges of metallic triangles with label I_1 which are connected to UTD surfaces or to a PEC ground plane (as specified with a BO or GF card). I_2 is not used.
- 3: Special microstrip line port load. The load is placed on all edges on the line between points (previously specified with DP cards) I_1 and I_2. A GF card with a conducting ground plane must be present.

R Real part of the complex load impedance.

X Imaginary part of the complex load impedance.

Note that the edge between triangles with labels I_1 and I_2 does not need to be straight. One may, for example, specify a resistive connection between two half cylinders.

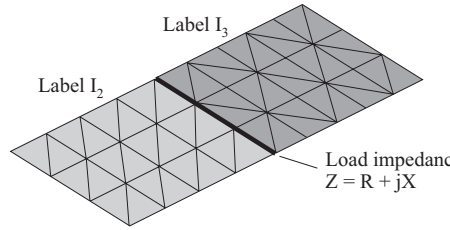


Figure 9-26: Application of the LE card

9.2.31 LP Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
LP		LAB					RP	LP	CP					
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

Here the discrete circuit elements in parallel, as shown in figure 9-27, can be assigned to a segment.

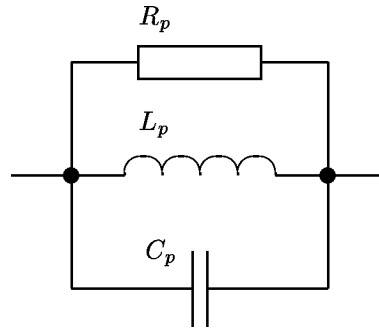


Figure 9-27: Sketch of the parallel circuit

Parameters:

LAB	All segments with this label are assigned the parallel circuit values specified below.
RP	Value of the resistor in Ω.
LP	Value of the inductor in H.
CP	Value of the capacitor in F.

The impedance is then given by

$$Z_p = \frac{1}{\frac{1}{R_p} + \frac{1}{j\omega L_p} + j\omega C_p}.$$

If RP=0, then the resistance is interpreted as infinite, i.e. in the parallel case it will not change the impedance. The same applies to LP.

The LP card may be combined with the LD, LS, LZ and the SK cards, but only one LP card may be used per label. If a second LP card is used, it replaces the values entered by the first one. This card has no significance for surface elements, even when these are assigned the same label.

9.2.32 LS Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
LS		LAB					RS	LS	CS					
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

Here the discrete circuit elements in series, as shown in figure 9-28, can be assigned to a segment.

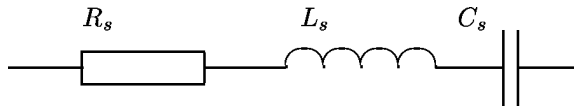


Figure 9-28: Sketch for the serial combination

Parameters:

LAB	All segments with this label will be assigned the elements in the series combination.
RS	Value of the resistor in Ω.
LS	Value of the inductor in H.
CS	Value of the capacitor in F.

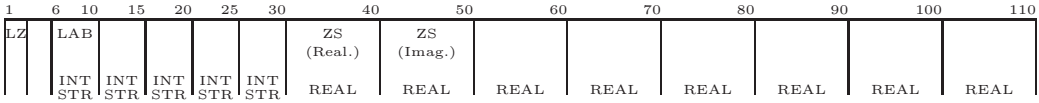
The impedance is given by

$$Z_s = R_s + j\omega L_s + \frac{1}{j\omega C_s}.$$

If CS=0 is selected, it is interpreted as infinite capacitance, i.e. in the case of the series combination it is zero.

The LS card may be combined with the LD, LP, LZ and the SK cards, but only one LS card may be used per label. If a second LS card is used, it replaces the values entered by the first one. This card has no significance for surface elements, even when these are assigned the same label.

9.2.33 LZ Card



Here a complex impedance can be assigned to a segment.

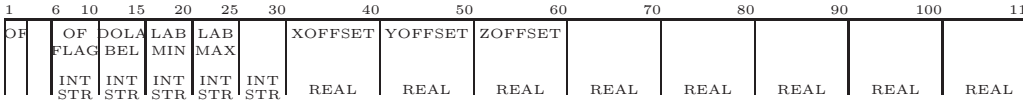
Parameters:

LAB	All segments with this label are assigned the impedance ZS.
ZS	The complex value of the impedance in Ω.

The value ZS is a constant, it is not dependent on the frequency. Frequency dependent impedances can be realised using the LS or the LP cards.

The LZ card may be combined with the LD, LP, LS and the SK cards, but only one LZ card may be used per label. If a second LZ card is used, it replaces the values entered by the first one. This card has no significance for surface elements, even when these are assigned the same label.

9.2.34 OF Card



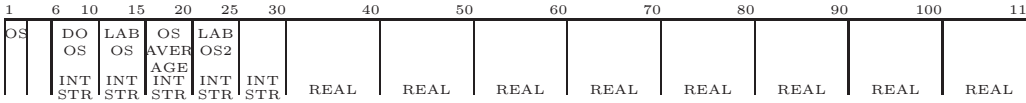
This card specifies an offset for the origin of the coordinate system used for near and far field calculations. In addition it is possible to use only a part of the structure when calculating the fields (selected using labels).

Parameters:

OFFLAG	0: No coordinate transformation, the fields are calculated in global coordinates.
	1: Use the offset specified below as the origin of the coordinate system for field calculations.
DOLABEL	0: All structures are used in the field calculation.
	1: Use label selection when calculating near and far fields. Only the currents on structures with a label in the range LABMIN...LABMAX are used during field computation. (If a basis function extends over, for example, two triangles it is included if either triangle lies in the specified range.)
LABMIN	Label range, see parameter DOLABEL.
LABMAX	Label range, see parameter DOLABEL.
XOFFSET	x coordinate of the transformed origin in m (it is scaled by the SF card if SKALFLAG=1).
YOFFSET	y coordinate of the transformed origin in m (it is scaled by the SF card if SKALFLAG=1).
ZOFFSET	z coordinate of the transformed origin in m (it is scaled by the SF card if SKALFLAG=1).

A possible application of the OF card is, for example, to calculate the near field on the surface of a sphere whose centre does not lie on the origin. The OF card transforms the origin of the coordinate system to the centre of the sphere, such that the near field calculation can be executed in spherical coordinates.

9.2.35 OS Card



With this card the currents on the surfaces and the segments can be extracted.

Parameters:

DOOS
- 0: No current output (but does start calculation).
- 1: Output all currents (segments and triangles).
- 2: Output all currents on triangles (metallic and dielectric).
- 3: Only output the currents on metallic segments.
- 4: Output all currents on segments and triangles with the label LABOS. (See also DOOS=7 which allows a range of labels.)
- 5: Export the currents on all segments to a `*.rsd` file in CableMod format (see the comment below).
- 6: Export the currents on all segments with labels in the range LABOS...LABOS2 to a `*.rsd` CableMod file.
- 7: Output the currents on all segments and triangles with labels in the range LABOS...LABOS2.

LABOS: Label for the selection when DOOS=4. If DOOS is set to 6 or 7 this is the start label.

OSAVERAGE
- 0: For the output of currents in the vertices of the triangles, neighbouring triangles with common vertices are identified. The current densities are then averaged over the neighbours. This ensures that the graphical representation is a smooth colour representation without discontinuities.
- 1: The averaging that may be very time consuming, particularly for structures containing a large number of triangles, are switched off.

LABOS2: When DOOS is 6 or 7 this specifies the end of the label range.

If the parameter DOOS=0 is specified no currents are extracted, but the solution is started if required.

With DOOS=4 only the currents associated with certain elements (specified with the label LABOS) can be extracted. If e.g. all the currents on the triangles or segments with the labels 0 and 4 are to be extracted, then the following two cards may be entered.

```
OS    4    0
OS    4    4
```

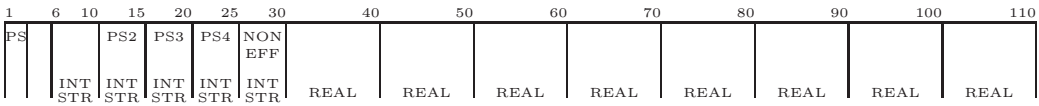
The options DOOS=5 and DOOS=6 permit the creation of a `*.rsd` file for use with the transmission line simulation program CableMod.[12] The currents along all (for DOOS=5) or selected segments (DOOS=6) are exported to the `*.rsd` file (the filename without extension is the same as that of the `*.fek` file). The `*.rsd` file is an ASCII file and contains first a description of the geometry of the line, followed by blocks with the current information for each frequency. It can be read by CableMod, and can also be imported back into FEKO to realise an impressed line source (see the AC card).

If the current of dielectric triangles (surface current formulation) must be output by the OS card, both the equivalent electric and magnetic surface currents of the external problem are written to the output file. (The currents of the internal problem are different to those of the external problem only in that their sign is reversed.)

If requested by the DA card, a `*.os` card will be created in addition to the currents written to the output file.

[12] To use the CableMod interface this module must be activated, if required please contact EMSS.

9.2.36 PS Card



This card is used to control program execution.

Parameters:

PS2
- 0: Normal execution.
- 1: The matrix elements are calculated and stored in a `*.mat` file.
- 2: The matrix elements are read from a `*.mat` file rather than calculated.
- 3: If a `*.mat` file exists from a previous calculation, the matrix is read from this file. Otherwise the matrix elements are calculated and saved in a `*.mat` file.

PS3
- 0: Normal execution.
- 1: The currents are calculated and stored in a `*.str` file.
- 2: The currents are read from the `*.str` file, i.e. the calculation of the matrix elements and the solution of the matrix equation are skipped.
- 3: If a file `*.str` exists from a previous calculation, the currents are read from this file. Otherwise the currents are calculated and saved in a `*.str` file for later use.

PS4 The expansion coefficients of the MoM basis functions are printed to the `*.out` output file:
- 0: All currents are printed.
- 1: No currents are printed (resulting in a shorter output file).
- 2: Currents on the MoM region.
- 4: Currents on node between segments.
- 8: Currents on a connection point.
- 16: Equivalent currents on the dielectric edges.
- 32: Polarisation currents in the dielectric cuboids.
- 64: Currents on the PO region.

Combinations are possible, e.g. with $12 = 8 + 4$ currents on both segments and connection points will be written to the output file. This output is not useful to general FEKO users, and it tends to make the `*.out` files very large. This option is therefore only supported in the superuser mode (see SU card).

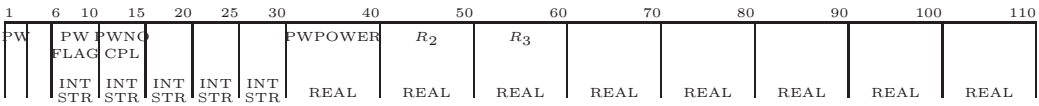
NONEFF
- 0: Normal matrix element calculation.
- 1: The calculation of the matrix elements as well as the field calculation are done in an inefficient way. This switch is only used to verify the result of the effective calculation. It is thus only used during program development.

This variable program control can lead to a large reduction in the computation time. When doing long calculations it is advisable to write the currents into a file by setting PS3=1 (or 3). If further calculations are then to be done, the currents can be read directly into memory and be processed by using the parameter PS3=2 (or 3), thus the matrix elements do not have to be recalculated and the matrix equation does not have to be resolved. The user must ensure that the geometry does not change between the two calculations — FEKO does not check for any differences.

It often happens that one, for example, runs a calculation on a workstation with large amounts of RAM and saves the currents in a `*.str` file. Then one may use a PC with less storage space to do further calculations (such as post processing). FEKO will try to allocate memory for the complete system matrix (out-of-core if required), but the currents will be imported from the `*.str` file.

If the parameter PS2 or PS3 is set, the PS card may occur only once in the input file. It is advisable to place it right after the EG card.

9.2.37 PW Card



When defining the excitation of an antenna the source is normally specified as a complex voltage. The PW card allows the user to specify the radiated power instead. In addition, it is possible to consider a mismatch between the antenna input impedance and the internal impedance of a voltage source or the characteristic impedance of a transmission line feed.

Parameters:

PWFLAG
- 0: PW card is not activated, i.e. the specified value U_0 of the voltage source is used.
- 1: PW card is activated and all the currents are multiplied by a scaling factor such that the total radiated power (the sum of the power radiated by all the individual sources) is P_0 (parameter PWPOWER). Mismatch is not considered.
- 2: All voltage sources are assumed to have an input impedance Z_i as specified by the parameters R_2 (real part) and R_3 (imaginary part). The currents are scaled such that the total power supplied by the voltage sources equals P_0 (parameter PWPOWER) as discussed below. The mismatch losses in the source impedance Z_i reduce the antenna gain.
- 3: All the antennas are assumed to be fed by transmission lines with a complex characteristic impedance Z_L as specified by the parameters R_2 (real part) and R_3 (imaginary part). If there is a mismatch between Z_L and the antenna input impedance Z_a, some of the incident power will be reflected back to the source. With PWFLAG=3 all currents are scaled such that the total incident power is P_0 (parameter PWPOWER) as discussed below. The reflected power reduces the antenna gain.

PWNOCPL
- 0: If multiple impressed sources (elementary dipoles A5/A6, impressed current elements AI/AV, etc.) are present, the mutual coupling of all these sources, as well as the coupling of the sources with other structures such as ground (BO card), UTD surfaces, or MoM elements is taken into account when determining the source power. This is also the default if the PW card is not present.

1: This mutual coupling is not considered. This is acceptable for sources which are relatively far away or when no accurate power values are required. (Since gain/directivity are based on power, they are then also possibly not very accurate.)

PWPOWER The total power P_0 in Watt, i.e. the total power supplied by all the voltage sources, or in the case of transmission lines, the total power of all forward travelling waves.

R_2 For PWFLAG=2: Real part of the source impedance Z_i.
For PWFLAG=3: Real part of the characteristic impedance Z_L of the transmission line.

R_3 For PWFLAG=2: Imaginary part of the source impedance Z_i.
For PWFLAG=3: Imaginary part of the characteristic impedance Z_L of the transmission line.

Details of the various possibilities with the use of the PW card are shown in figure 9-29.

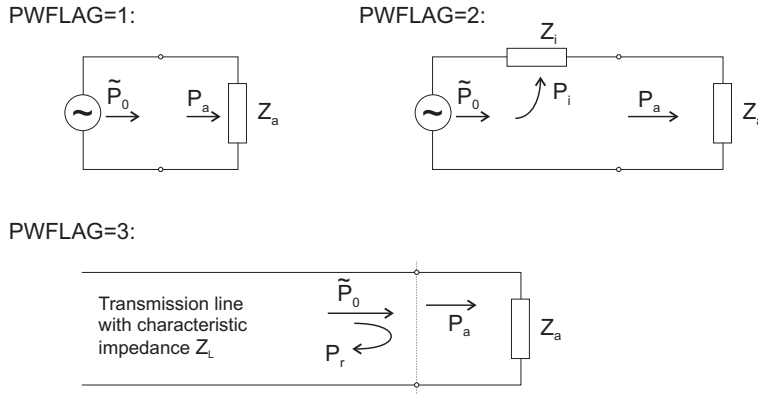


Figure 9-29: Possible applications of the PW card to determine the total power

The options PWFLAG=2 and PWFLAG=3 are only allowed for voltage sources (the A1, A2, A3, A4, A7 and AE cards). For models containing other sources such as dipoles and impressed currents, the option PWFLAG=1 should be used. For plane waves PWFLAG must be 0.

The power equations for different cases are discussed below. Consider, in general, that there are N sources (such as in an array antenna) with open circuit voltages $U_{0,\nu}$ (before the scaling operation) where the parameter ν lies in the range $1 \ldots N$. At each source there is an antenna input impedance $Z_{a,\nu}$ (as calculated during the FEKO solution) to which power $P_{a,\nu}$ is transferred.

- PWFLAG=1:

  When PWFLAG=1 all the source power is delivered to the respective antennas, i.e.

  $$\tilde{P}_{0,\nu} = P_{a,\nu}$$

  as shown in figure 9-29. Thus the total power can be scaled to P_0 with the factor

  $$s = \frac{P_0}{\sum_{\nu=1}^{N} \tilde{P}_{0,\nu}} = \frac{P_0}{\sum_{\nu=1}^{N} P_{a,\nu}}$$

  The currents on the structure are scaled with the factor $\sqrt{s}$. There is no power loss.

- PWFLAG=2:

  When PWFLAG=2 the internal impedance Z_i of the voltage source is considered as shown in figure 9-29. Since the same current flows through the internal source impedance and the antenna input impedance, the power dissipated in the impedance of the νth voltage source is given by the relation

  $$P_{i,\nu} = P_{a,\nu} \frac{\operatorname{Re} Z_i}{\operatorname{Re} Z_{a,\nu}}$$

  and the scaling factor s (to scale the total power supplied by the sources to P_0) is

  $$s = \frac{P_0}{\sum_{\nu=1}^{N} \tilde{P}_{0,\nu}} = \frac{P_0}{\sum_{\nu=1}^{N} (P_{a,\nu} + P_{i,\nu})} = \frac{P_0}{\sum_{\nu=1}^{N} P_{a,\nu} \left(1 + \frac{\operatorname{Re} Z_i}{\operatorname{Re} Z_{a,\nu}}\right)}$$

  The combined loss caused by the mismatched antennas

  $$P_{\text{loss}} = s \sum_{\nu=1}^{N} P_{i,\nu} = s \sum_{\nu=1}^{N} P_{a,\nu} \frac{\operatorname{Re} Z_i}{\operatorname{Re} Z_{a,\nu}}$$

  reduces, for example, the antenna gain (but not the directivity).

- PWFLAG=3:

  When PWFLAG=3 each antenna (with input impedance $Z_{a,\nu}$) is considered to be excited by a transmission line with a complex characteristic impedance Z_L as shown in figure 9-29. The reflection factor

  $$\varrho_\nu = \frac{Z_{a,\nu} - Z_L}{Z_{a,\nu} + Z_L}$$

  is taken into account when calculating the incident power at the feed point.

The total incident power is given by

$$\tilde{P}_{0,\nu} = \frac{P_{a,\nu}}{1 - |\varrho_\nu|^2}$$

and the reflected power by

$$P_{r,\nu} = |\varrho_\nu|^2 \, \tilde{P}_{0,\nu} = P_{a,\nu} \, \frac{|\varrho_\nu|^2}{1 - |\varrho_\nu|^2}$$

To ensure that the total incident power is P_0, the power is scaled with the factor

$$s = \frac{P_0}{\sum_{\nu=1}^{N} \tilde{P}_{0,\nu}} = \frac{P_0}{\sum_{\nu=1}^{N} \frac{P_{a,\nu}}{1 - |\varrho_\nu|^2}}$$

and the currents with the factor $\sqrt{s}$. As before the total reflected power

$$P_{\text{loss}} = s \sum_{\nu=1}^{N} P_{r,\nu} = s \sum_{\nu=1}^{N} P_{a,\nu} \, \frac{|\varrho_\nu|^2}{1 - |\varrho_\nu|^2}$$

reduces the gain of the antenna.

9.2.38 SK Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
SK		I_1	I_2	I_3			R_1	R_2	R_3	R_4	R_5	R_6	R_7	
							R'_1	R'_2	R'_3	R'_4	R'_5	R'_6	R'_7	R'_8
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

Using this card the "skin effect" and ohmic losses can be examined on wire segments and surface elements. In addition it is possible to switch from metallic triangles to thin dielectric layers (possibly consisting of multiple layers).

Parameters:

- I_1 This card affects all segments and surface elements with label I_1.
- I_2 0: Ideal conductivity is assumed (also the default when there is no SK card for a given label). All other parameters are ignored.
  1: Use the high frequency skin effect approximation.
  2: Use the static approximation of the skin effect (ohmic losses).
  3: Use the exact expression of the skin effect for wires and/or surfaces that is valid at arbitrary frequencies.
  4: Treat metallic triangles as thin isotropic dielectric layers (possibly consisting of multiple layers).
  5: Treat metallic triangles as thin anisotropic dielectric layers (possibly consisting of multiple layers).

for $I_2 = 1$, 2 or 3:

- R_1 The thickness d of the surface elements in m (if an SF card is present, this *is scaled* independent of the value of SKALFLAG).
- R_2 The relative permeability μ_r of the material.
- R_3 The conductivity σ in $\frac{1}{\Omega \mathrm{m}}$.
- R_7 Magnetic loss factor $\tan\delta_\mu$ (the complex permeability is then given by $\mu = \mu_0\mu_r\left(1 - j\tan\delta_\mu\right)$).

for $I_2 = 4$:

- I_3 The number of layers. If this field is empty, 0 or 1 there is just one layer.
- R_1 The thickness d of the first layer in m (if an SF card is present, this *is scaled* independent of the value of SKALFLAG).
- R_2 The relative permeability μ_r. (All layers must have the same permeability.)
- R_3 The conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ of the first layer.
- R_5 The loss factor $\tan\delta$ ($\tan\delta = \frac{\sigma}{\omega\varepsilon_0\varepsilon_r}$) of the first layer.
- R_6 The relative dielectric constant ε_r of the first layer.

R_7 Magnetic loss factor $\tan\delta_\mu$ (the complex permeability is then given by $\mu = \mu_0\mu_r\,(1 - j\tan\delta_\mu)$). (All layers must have the same permeability.)

now follow $I_3 - 1$ lines with the parameters of the remaining layers (this approximation is such that it is independent of the order of the layers)

R'_1 The thickness d of the nth layer in m (if an SF card is present, this *is scaled* independent of the value of SKALFLAG).

R'_3 The conductivity σ in $\frac{1}{\Omega\text{m}}$ of the nth layer.

R'_5 The loss factor $\tan\delta$ ($\tan\delta = \frac{\sigma}{\omega\varepsilon_0\varepsilon_r}$) of the nth layer.

R'_6 The relative dielectric constant ε_r of the nth layer.

for $I_2 = 5$:

I_3 The number of layers. If this field is empty, 0 or 1 there is just one layer.

R_2 The relative permeability μ_r. Note that all layers must have the same permeability.

R_3 x component of the vector ζ (used to define the principle direction, see R'_2 below).

R_4 y component of the vector ζ.

R_5 z component of the vector ζ.

R_7 Magnetic loss factor $\tan\delta_\mu$ (the complex permeability is then given by $\mu = \mu_0\mu_r\,(1 - j\tan\delta_\mu)$). Note that all layers must have the same permeability.

now follow I_3 lines with the parameters of the layers (this approximation is such that it is independent of the order of the layers)

R'_1 The thickness d of this layer in m (if an SF card is present, this *is scaled* independent of the value of SKALFLAG).

R'_2 The angle α_i (in degrees) by which the reference (principle) direction of this layer is rotated with respect of the projection of the vector ζ onto the triangles.

R'_3 Conductivity σ in ($\frac{1}{\Omega\text{m}}$) of this layer in the principal direction.

R'_4 Conductivity σ in ($\frac{1}{\Omega\text{m}}$) of this layer in the orthogonal direction.

R'_5 The loss factor $\tan\delta$ ($\tan\delta = \frac{\sigma}{\omega\varepsilon_0\varepsilon_r}$) of this layer in the principal direction.

R'_6 The relative dielectric constant ε_r of this layer in the principal direction.

R'_7 The loss factor $\tan\delta$ ($\tan\delta = \frac{\sigma}{\omega\varepsilon_0\varepsilon_r}$) of this layer in the orthogonal direction.

R'_8 The relative dielectric constant ε_r of this layer in the orthogonal direction.

Having both triangles and segments with the label I_1 should be avoided. Separate labels and a distinct SK card for each label should be used. In addition all wires with the label I_1 must have the same radius. If this is not the case a unique label must be introduced for each radius.

The application of the card depends on the value of I_2. It should be noted that the skin depth is given by $\delta_{skin} = \sqrt{\frac{2}{\omega\mu\sigma}}$, where the radial frequency $\omega = 2\pi f$ and the permeability $\mu = \mu_r\mu_0$.

- I_2=0:
  No further parameters.
- I_2=1 (high frequency skin effect approximation):
  The required parameters are μ_r, $\tan\delta_\mu$ and σ, and for surfaces also the thickness d. A good conductivity is required, i.e. $\sigma \gg \omega\varepsilon_0$.
  - For wires a further condition requires that $\delta_{skin} < \varrho$ where ϱ is the wire radius. The surface impedance is given by $Z_s' = \frac{1}{2\pi\varrho}\sqrt{\frac{j\omega\mu}{\sigma}}$.
  - For metallic surfaces the condition $\delta_{skin} < \frac{d}{2}$ must be met. The surface impedance is given by $Z_s = \frac{1}{2}\sqrt{\frac{j\omega\mu}{\sigma}}$.
- I_2=2 (ohmic losses):
  The required parameters are μ_r, $\tan\delta_\mu$ and σ, and for surfaces also the thickness d. A good conductivity is required, i.e. $\sigma \gg \omega\varepsilon_0$.
  - For wires a further condition requires that $\delta_{skin} > \varrho$ where ϱ is the wire radius. The surface impedance is given by $Z_s' = \frac{1}{\pi\varrho^2}\frac{1}{\sigma}$.
  - For metallic surfaces the condition $\delta_{skin} > \frac{d}{2}$ must be met. The surface impedance is given by $Z_s = \frac{1}{\sigma d}$.
- I_2=3 (exact expression of the skin effect):
  The required parameters are μ_r, $\tan\delta_\mu$ and σ, and for surfaces also the thickness d. A good conductivity is required, i.e. $\sigma \gg \omega\varepsilon_0$.
  - For wires with radius ϱ the surface impedance is given by

    $$Z_s' = \frac{1-j}{2\pi\varrho\sigma\delta_{skin}} \frac{J_0\left[(1-j)\frac{\varrho}{\delta_{skin}}\right]}{J_1\left[(1-j)\frac{\varrho}{\delta_{skin}}\right]}$$

    where J_0 and J_1 are Bessel functions.
  - For metallic surfaces the surface impedance is given by

    $$Z_s = \frac{1-j}{2\sigma\delta_{skin}} \frac{1}{\tan\left[(1-j)\frac{d}{2\delta_{skin}}\right]}$$

Examples are given in `example_02` and `example_33` in the *Examples Guide*.

- I_2=4 (thin isotropic dielectric layers):
  This option only makes sense for triangular surfaces, not for wires. The required parameters are d, μ_r, $\tan\delta_\mu$ and ε_r as well as σ or $\tan\delta$. Under special circumstances it is possible to specify both σ and $\tan\delta$ (for example, to approximate a specific frequency response). FEKO will give a warning which may be ignored.

  The triangles with the label I_1 exist in a certain environment ε_e, μ_e, which is usually specified with the EG card. It is also possible to place the triangles within a dielectric body — in this case the environment is specified by the parameters of the DI card. The use of I_2=4 requires that $\mu = \mu_e$ and $\varepsilon \neq \varepsilon_e$ where $\mu = \mu_r\mu_0$ and the complex dielectric constant $\varepsilon = \varepsilon_r\varepsilon_0\,(1 - j\tan\delta) - j\frac{\sigma}{\omega}$ (normally either σ or $\tan\delta$ is entered as zero). A further condition is that the triangle should be geometrically thin, i.e. d must be small relative to the lateral dimensions. The mesh size is determined by the wavelength in the environment (i.e. in the medium ε_e, μ_e).

  For a single layer, the card consists of only one line. The surface impedance, as used by FEKO, is then

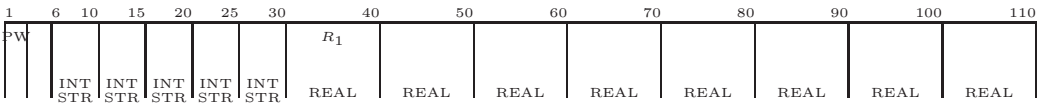
$$Z_s = \frac{\beta}{2j\omega\,(\varepsilon - \varepsilon_e)\,\tan(\beta\frac{d}{2})}$$

  where $\beta = \omega\sqrt{\varepsilon\mu}$ is the propagation constant. An example is given in `example_32` (*Examples Guide*). For multiple lines the card required one line per layer with the parameters of the first layer on the same line as the card name. The approximate surface impedances of the different layers are added to determine the effective surface impedance.

- I_2=5 (thin anisotropic dielectric layers):
  This option is very similar to I_2=4, but the layers are anisotropic. The principle direction in each layer is defined by the angle α relative to the projection of the vector ζ onto the plane of triangle. Here α is measured in the mathematically positive sense with respect to the normal vector of the triangle. (WinFEKO can be used to display the fibre direction and visually check that the input file is correct.) In this case the card line is followed by an additional line for each layer.

If no SK card has been defined, FEKO assumes ideal conductivity without any losses.

9.2.39 SP Card



This card is used to calculate the S-parameters for the active sources.

Parameters:

R_1 The reference impedance. This is used for all sources for which no impedance value is specified when defining the source. If this field is empty, it defaults to 50 Ω.

The N ports must be defined before using the SP card. They are identified simply by defining excitation cards. Currently only A1, A2, A3, A4 and AE sources are supported. A1, A2 and A3 sources must be selected by label (not with position), and unique labels must be used (i.e. no other segments or triangles may have a label which is used for a port).

If the amplitude of any port is set to zero, it will be used as a receive port (or sink) but not as a source. For example, if only S_{21} and S_{11} are required for a two port network, one may set the amplitude of the source defining port 2 to exactly zero. Then S_{12} and S_{22} are not calculated — in some cases this may save considerable computation time.

The load impedance for each of the port sources can be specified at the source itself. If no such impedance was specified, the R_1 value specified with the SP card will be used (if this value is not specified it defaults to 50 Ω). This load impedance will be added automatically to each port.

It must be noted that the SP card adds load impedances to all the ports. (For A1, A2 and A3 sources it uses LZ type loads; for A4 sources it uses L4 type loads and for AE sources it uses LE type loads.) If any similar loads were applied to the source position before the SP card these loads will be overwritten. Also when execution continues after processing the SP card these loads will still be present. The original amplitudes of the excitations will, however, be restored.

9.2.40 TL Card

1	6	10	15	20	25	30	40	50	60	70	80	90	100	110
TL		I_1	I_2	I_3	I_4	I_5	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
							R'_1	R'_2	R'_3	R'_4	R'_5	R'_6		
		INT STR	INT STR	INT STR	INT STR	INT STR	REAL	REAL	REAL	REAL	REAL	REAL	REAL	REAL

This card is used to connect a non-radiating transmission line between two segments.

Parameters:

I_1 -1: This TL card does not define a transmission line, but all previously defined transmission lines are deleted. All the other input parameters are ignored.

0: Defines a new transmission line, all previously defined transmission lines are replaced.

1: An additional transmission line is defined.

I_2 Label of the segment which represents the start of the transmission line. If more than one segment with this label exists, then the last segment with this label is used. In the special case $I_2 = -1$, the start point of the transmission line is defined by specifying its Cartesian coordinates (R'_1, R'_2 and R'_3) in a second line immediately following the TL card. (These coordiantes must be at the centre of a wire segment.)

I_3 Label of the segment which represents the end of the transmission line. If more than one segment with this label exists, then the last segment with this label is used. In the special case $I_3 = -1$, the end point of the transmission line is defined by specifying its Cartesian coordinates (R'_4, R'_5 and R'_6) in a second line immediately following the TL card. (These coordiantes must be at the centre of a wire segment.)

I_4 The positive port voltage definition is according to the direction of the segment that it is connected to (from the start to the end point of the segment). Thus the input and output ports of the transmission line have unique orientations. For $I_4 = 0$ (or empty) the transmission line connecting the ports is not crossed, for $I_4 = 1$, it is crossed.

R_1 The length of the transmission line in metres. In the special case $R_1 = -1$, FEKO determines the length based on the geometrical distance between the start and end points. R_1 is scaled with the scaling factor of the SF card.

R_2 Real part of the characteristic impedance of the transmission line in Ohm.

R_3 Imaginary part of the characteristic impedance of the transmission line in Ohm. Note that the characteristic impedance only defines the ratio between the voltage and current of the two waves propagating along the line. It does not specify any losses.

R_4 Real part of the shunt admittance at port 1 in Siemens. (This admittance is across the port, connecting the two wires of the transmission line.)

R_5 Imaginary part of the shunt admittance at port 1 in Siemens.

R_6 Real part of the shunt admittance at port 2 in Siemens.

R_7 Imaginary part of the shunt admittance at port 2 in Siemens.

R_8 Losses of the transmission line in dB/m. Note that since the propagation constant is taken as the propagation constant of the medium in which the start and end segments are located, the attenuation specified by R_8 is added to any losses of this medium. This factor is not affected by the scaling factor specified with the SF card, i.e. if a scaling factor which reduces the length of the transmission line is added, the total loss through the line will be less as the loss is still R_8 in dB/m.

For $I_2 = -1$ or $I_3 = -1$ follows a second line:

R_1 x coordinate of the centre of the port 1 segment in metres. (The scaling factor of SF card applies.)

R_2 y coordinate of the centre of the port 1 segment.

R_3 z coordinate of the centre of the port 1 segment.

R_4 x coordinate of the centre of the port 2 segment.

R_5 y coordinate of the centre of the port 2 segment.

R_6 z coordinate of the centre of the port 2 segment.

An arbitrary number of transmission lines can be used, also one wire segment could be for instance the end of one transmission line and the start of another. All transmission lines ending at one wire are connected in parallel.

Any load impedance defined over the transmission line port segments with the LZ, LS, LP, LD, CO or SK cards are placed in series with the port, parallel admittances can be defined directly at the TL card.

If a voltage source of type A1 or A3 is applied at one of the port segments, then this voltage source is assumed to be across the port (i.e. feeding the transmission line directly with an impressed voltage). Any other sources are in series with the port.

FEKO automatically determines the type of the transmission line network:

1 no impressed voltage source at both ports
2 impressed voltage source at port 1
3 impressed voltage source at port 2
4 impressed voltage source at both ports

Both wire segments for port 1 and port 2 should be located in the same medium, so that the propagation constant of this medium can be taken for the transmission line. If the segments are in different media, then the medium of the segment at port 1 is used. Note that the propagation constant and thus also the propagation loss of the transmission line is the same as that of the medium surrounding port 1 unless an additional loss factor is specified with R_8. If this is free space the transmission line will be lossless. For transmission lines with a propagation constant that is higher than that of the surrounding medium, such as coaxial cables filled with dielectric material, one needs to reduce the length of the transmission line.

Losses in the transmission line network (due to the shunt admittances or transmission line losses directly) are taken into account and will for instance reduce antenna efficiency or gain.

The TL card is used in `example_39` (*Examples Guide*) to create a log periodic antenna.

10 The optimiser OPTFEKO

10.1 Description

With optimising, certain properties should be improved by changing a number of specified parameters. For example, the gain property of an antenna can be optimised by varying the parameters of the geometry. Certain parameters are allowed to be varied within certain bounds and are called the optimising parameters. As discussed in section 6.3 the program PREFEKO can handle symbolic variables. Therefore the input file `*.pre` can be created using symbolic variables to represent the optimisation parameters.

Apart from defining the optimisation parameters the aim (or goal) of the optimisation must be specified. An aim function is used to determine how well the current solution approximates the desired goal. The aim function can, for example, be defined such that its minimum is the optimum solution. A number of different optimisation methods are available with OPTFEKO.

The program OPTFEKO requires two input files from the user.

10.2 The *.pre input file

The first required file is the normal `*.pre` input file for PREFEKO, where the symbolic optimisation parameters are used to define the variables that must be optimised. During optimisation these parameters are varied by OPTFEKO. It repeatedly generates new `*.pre` files, in which the optimisation parameters are assigned. The user can assign values to these variables (for example in order to run PREFEKO and view the geometry) using the DEFINED function. (An example is given in section 10.5).

10.3 The *.opt input file

The optimisation method, its parameters and the aim of the optimisation are stored in a second file with a `*.opt` extension. This file consists of three or four sections:

- Assignment of optimisation parameters, minimum and maximum values
- Optional assignment of the penalty function
- Set optimisation parameters (e.g. step size, final value)
- Set the aim function with the required parameters

In the `*.opt` file blank and/or comment lines (starting with `**`) are allowed between the sections, but they are not allowed to appear in the sections themselves. The parameters

are space delimited, i.e. they are separated by spaces and can be placed in any column, but they must be entered in the correct order.

The keywords used in the `*.opt` file exist in both German and English (for example, `RASTERSUCHE` and `GRID_SEARCH`). Both versions of each keyword will be given in the discussion below. OPTFEKO will recognise keywords in either language, independent of the language selected by the environment variable `FEKOLANG`.

A documentation file `example.opt` containing more detail is provided with the FEKO installation (in the `doc\optfeko` subdirectory). The file is also located in the same directory on the FEKO CD.

10.3.1 Definition of optimisation parameters

The optimisation parameters are defined in tabular form. Each optimisation parameter has a name (a symbolic variable used, but not numerically defined, in the `*.pre` file). For each parameter a start value as well as a minimum and maximum value have to be given.

Example:

```
** The optimisation parameters follow:
** Name   Begin value     Minimum        Maximum
#alpha          20          -80             80
#a            0.75         0.25            2.0
```

Normally the minimum and maximum values of the optimisation ** parameters as specified in the `*.opt` file are only used for the normalisation of the parameter space. In order to ensure that the parameters stay within certain limits, a penalty function (see section 10.3.2) can be added. This will result in a smooth aim function for the optimisation.

In some circumstances, however, sharp boundaries must be enforced to avoid invalid geometries (for example, the distance between points cannot become negative). If the keyword `ERZWINGE_MIN_MAX` or `ENFORCE_MIN_MAX` is found in the `*.opt` file, then the provided boundaries are strictly enforced when OPTFEKO creates a `*.pre` file for FEKO to ensure its validity. OPTFEKO internally still assumes that, for example, the obtained input impedance is for the original set of parameters, i.e. one should still add a penalty function in order to force the optimisation algorithms to move back into the valid parameter range.

10.3.2 Definition of the penalty function

To ensure that the optimisation parameter x stays within the bounds x_{min} and x_{max}, a penalty function can be added to the aim function. The penalty function P is defined by

the equation

$$P = \begin{cases} 10 \cdot P_u \cdot \frac{x_{min}-x}{x_{max}-x_{min}} & \text{for} \quad x < x_{min} \\ 0 & \text{for} \quad x_{min} \leq x \leq x_{max} \\ 10 \cdot P_o \cdot \frac{x-x_{max}}{x_{max}-x_{min}} & \text{for} \quad x > x_{max} \end{cases}$$

The two parameters P_u and P_o define the value of the penalty function when overestimating the allowable value by 10%. When the penalty function is added to the aim function, it must be ensured that they both are of the same dimension.

The penalty function is defined with the keyword `PENALTYFUNKTION` or `PENALTY_FUNCTION`. In the rows that follow, the optimisation parameters P_u and P_o are specified. P_u and P_o in the first row are associated with the first parameter in the optimisation parameters section. P_u and P_o in the following rows are associated with the optimisation parameters defined in the corresponding rows. If no penalty function should be assigned to a specific optimisation parameter, $P_u = P_o = 0$ should be set. If no penalty functions are to be assigned at all, then the whole section can be left out.

Example:

```
PENALTY_FUNCTION
    100      100
     10       10
```

10.3.3 Definition of the optimisation process

There are a number of optimisation processes that can be used with OPTFEKO.

Discrete points

This method is strictly speaking not an optimisation method. The optimisation parameters are linearly varied between their minimum and the maximum values. The value of the aim function is calculated at n discrete points. The process is selected using the keyword `RASTERSUCHE` or `GRID_SEARCH`. The number of discrete points required for each optimisation parameter is specified in the line following the keyword.

Example:

```
GRID_SEARCH
  8   9
```

In this case the aim function will be determined $8 \cdot 9 = 72$ times. (Eight discrete points for the first optimisation parameter and nine discrete points for the second optimisation parameter).

Simplex method

Here the so-called Simplex is assigned through space over the normalised ([0 . . . 1]) optimisation parameters and the minimum is searched for.

In the row that follows the keyword `SIMPLEXVERFAHREN` or `SIMPLEX_METHOD`, four parameters need to be specified. The first indicates the basis of the Simplex in the normalised space. If the Simplex starts rotating around a minimum, its basis will be scaled down by the factor given by the second parameter. The third parameter is the convergence criteria. If the basis of the Simplex is smaller than this value, then the optimisation is stopped. The fourth parameter is also a convergence criteria. If the standard deviation of the values of the aim function in the corner points of the Simplex is smaller than this value, then the optimisation is also stopped.

Example:

```
SIMPLEX_METHOD
  0.15    0.5  0.001  0.001
```

The simplex method by Nelder Mead

This method is similar to the above-mentioned method, but it is implemented according to the Nelder Mead variation.

In this case seven parameters are specified after the keyword `SIMPLEXVERFAHREN` or `SIMPLEX_METHOD`. The first indicates the basis of the Simplex in the normalised space. If the Simplex starts rotating around a minimum, its basis will be scaled down by the factor given by the second parameter. The third parameter is the convergence criterion. If the basis of the Simplex is smaller than this value, then the optimisation is stopped. The fourth parameter is also a convergence criterion. If the standard deviation of the values of the aim function in the corner points of the Simplex is smaller than this value, then the optimisation is also stopped.

The last three parameters represent reflection, contraction and expansion. With contraction the Simplex is changed similar to reflection, but becomes either smaller (negative contraction) or somewhat larger (positive contraction) than its original size. Contraction is executed when normal reflection gives a node whose aim function is not smaller than the value of the Simplex node with the second highest value of the aim function. Expansion stretches the Simplex beyond the range of a normal reflection. It only takes place when the value of the aim function on the normally reflected node is smaller than that on the node with the previously smallest value of the aim function (the output Simplex).

Example:

```
SIMPLEX_METHOD
  0.15    0.5  0.001  0.001  1.0  0.5  2.0
```

Conjugate gradient method

The keyword for this optimisation method is `KONJUGIERTE_GRADIENTEN_VERFAHREN` or `CONJUGATE_GRADIENT_METHOD`. A full description of the conjugate gradient variants implemented in OPTFEKO (Fletcher-Reeves or Polak-Ribiere) and their necessary parameters can be obtained from the file `example.opt`.

Example:

```
CONJUGATE_GRADIENT_METHOD
    2  0.1  1.0E-6  1.0E-4  1.0E-3  1.618034  100.0   1.0  2  0  0.01
```

Quasi-Newton method

The keyword used for this method is `QUASI_NEWTON_VERFAHREN` or `QUASI_NEWTON_METHOD`. A full description of the Quasi-Newton methods — Davidon-Fletcher-Powell (DFP) and Broyden-Fletcher-Goldfarb-Shanno (BFGS) — can by found in the file `example.opt`.

Example:

```
QUASI_NEWTON_METHOD
2  0.1  1.0E-15  2.5E10  1.0E-4  1.0  100.0  1.0E-7  1.0E-4  2  0.01
```

For more information on the optimisation parameters please consult the file `example.opt` in the `doc\optfeko` subdirectory under the FEKO installation directory. (It is also available in this directory on the FEKO CD.)

10.3.4 Defining the aim function

A number of different aim functions are available in OPTFEKO.

Gain

With this aim function a maximisation of the gain/directivity of an antenna in one or more directions can be done. The optimisation can be done over a broad band by examining a number of frequencies. In such a case the *average* gain is maximised.

This aim function is selected by using the keyword `GEWINN` or `GAIN` in the `*.opt` file. In the next row, two additional parameters (M and N) are specified. M is a flag with the following meaning:

$$M = \begin{cases} 0 & \text{both polarisations} \\ 1 & \text{horizontal polarisation} \\ 2 & \text{vertical polarisation} \end{cases}$$

N indicates the number of blocks in the output file from FEKO, that are to be read. For each block $i = 1 \ldots N$ the first row is read and the gain/directivity g_i in dB is extracted. The aim function Z is then defined by

$$Z = -\frac{1}{N} \cdot \sum_{i=1}^{N} g_i$$

The negative sign ensures that a minimisation of the aim function maximises the gain.

Apart from the value of the aim function Z the standard deviation of the g_i with $i = 1 \ldots N$, the block number i of the block with the worst value of the aim function and the value of this aim function are given.

Example:

```
GAIN
  0   1
```

Isotropic radiators

This aim function allows the user to design an antenna with the best possible isotropic gain. Here one can optimise for more than one frequency by using the frequency blocks from the output file from FEKO in the aim function.

The parameters M, N_B and N_S are all specified on the line that follows the key word `RUNDSTRAHLUNG` or `OMNIDIRECTIVITY`. M is a flag that has the following meaning:

$$M = \begin{cases} 1 & E_\vartheta \quad \text{components used} \\ 2 & E_\varphi \quad \text{components used} \end{cases}$$

N_B is the number of the far field data blocks that are to be read and N_S indicates the number of rows that are to be read from each block, i.e. the number of discrete points (ϑ_j, φ_j) when varying the angle.

For each block $i = 1 \ldots N_B$ the roughness in dB is determined according to the relation

$$f_i = 20 \cdot \log_{10} \left[\frac{\max_j |E_{\vartheta,\varphi}(\vartheta_j, \varphi_j)|}{\min_j |E_{\vartheta,\varphi}(\vartheta_j, \varphi_j)|} \right].$$

Here j scans through the values $1 \ldots N_S$. The aim function Z is derived from

$$Z = \frac{1}{N_B} \cdot \sum_{i=1}^{N_B} f_i,$$

i.e. it is an average roughness in dB.

Apart from the value of the aim function Z, the standard deviation of f_i with $i = 1 \ldots N_B$, the block number i of the block with the worst value of the aim function and the current value of the aim function are written to the log file.

Example:

```
OMNIDIRECTIVITY
  2  10  90
```

Radiation pattern

This aim function can be used to optimise the radiation pattern according to an arbitrary shape specified by the user. As for the isotropic radiator one can optimise for more than one frequency by using the frequency blocks from the output file from FEKO in the aim function.

The parameters M, N_B, N_S and N_P are all specified on the line that follows the key word `RICHTDIAGRAMM` or `RADIATION_PATTERN`. M is a flag that has the following meaning:

$$M = \begin{cases} 1 & E_\vartheta \quad \text{components used} \\ 2 & E_\varphi \quad \text{components used} \end{cases}$$

N_B is the number of the far field data blocks that are to be read and N_S indicates the number of rows that are to be read from each block, i.e. the number of discrete points (ϑ_j, φ_j) when varying the angle. N_P gives the number of lines to follow with the definition of the aim radiation pattern. This is followed by N_P lines of the form

$N \quad VALUE$

where N is an index and $VALUE$ is the value of the normalised radiation pattern for this angle N. (N must be in the range $1 \ldots N_S$ and $VALUE$ in the range $0 \ldots 1$ at it is a normalised quantity.)

The aim function which is used by FEKO is computed as follows

$$Z = \frac{1}{N_B} \frac{1}{N_S} \sum_{i=1}^{N_B} \sum_{j=1} N_S \left| \frac{|E_{\vartheta,\varphi}(\vartheta_j^i, \varphi_j^i)|}{E_{\max}^i} - P(j) \right|$$

where $P(j)$ is the user specified aim radiation pattern with j is in the range $1 \ldots N_S$. OPTFEKO automatically uses a linear interpolation for points in between the N_P specified points. Should $P(1)$ not be defined by the user, OPTFEKO assumes $P(1) = 0$.

$E_{\vartheta,\varphi}(\vartheta_j^i, \varphi_j^i)$ is the electric field strength (the component which is selected by M) for the jth far field line in the ith block.

$E_{\max}^i$ represents the maximum field strength amplitude (the component which is selected by M) in the ith far field block, i.e. $E_{\max}^i = \max_j |E_{\vartheta,\varphi}(\vartheta_j^i, \varphi_j^i)|$ which is required to normalise the field strength to give the normalised radiation pattern.

For example, if one computes the full horizontal cut of the radiation pattern for an antenna to be optimised (vertical polarisation) and at the FF card in FEKO computes the field

for $\vartheta = 90°$ and $\varphi = 10°$, $20°$, $30°$, ... $360°$ ($\varphi = 0°$ has been excluded since it is equal to $\varphi = 360°$), the values would be $M = 1$ (vertical pol), $N_B = 1$ and $N_S = 36$. If one wants to design, for example, a sector radiator which gives zero radiation in the angular ranges $\varphi = 0° \ldots 90°$ and $\varphi = 270° \ldots 360°$ and maximum radiation in the angular range $\varphi = 150° \ldots 210°$ (and a linear increase/decrease in the angular ranges $\varphi = 90° \ldots 150°$ and $\varphi = 210° \ldots 270°$), the definition in the `*.opt` file would be

```
RADIATION_PATTERN
    1           1           36          6
    1           0.0
    9           0.0
   15           1.0
   21           1.0
   27           0.0
   36           0.0
```

This can be used, for example, to determine the amplitude and phase of four impressed Hertzian dipoles to get the best possible approximation of the pattern discussed above. The `*.pre` file will then contain something like

```
#l = 0.3*#lambda
A5   0                      #a1       #p1       #l        0         0         0
A5   1                      #a2       #p2       0         #l        0         0
A5   1                      #a3       #p3       -#l       0         0         0
A5   1                      #a4       #p4       0         -#l       0         0

** Compute the horizontal radiation pattern (10 deg. stepping)
FF   1    1    36   0         90        0                   10
```

and the `*.opt` file will contain a variable section giving ranges for these eight variables, an optimisation method and the `RADIATION_PATTERN` block given above. See also the files `pattern.opt` and `pattern.pre` in the subdirectory `examples\utils\optfeko` under the FEKO installation directory.

Radar cross section (RCS)

With this aim function the RCS σ can be minimised or maximised (either the average value or the minimum / maximum value) over a range of angles and over a range of frequencies (blocks).

This aim function is defined by the keyword `RCS` (it is the same in German and English). The parameters M, N_B and N_S are specified on the next line. N_B specifies the number of blocks that will be considered, and N_S the number of values read from each block, for example the number of angles for which the calculation is to be made. The parameter M

has the following meaning:

$$M = \begin{cases} -2 & \text{Maximise the minimum RCS} \\ -1 & \text{Maximise the average RCS} \\ 1 & \text{Minimise the average RCS} \\ 2 & \text{Minimise the maximum RCS} \end{cases}$$

The average value or minimal / maximal value of the RCS is determined from the $N_B \times N_S$ calculated values. In the case of the average value, the aim function Z, is given by the following relation

$$Z = \text{sign}(M) \cdot \frac{1}{N_B} \cdot \frac{1}{N_S} \cdot \sum_{i=1}^{N_B} \sum_{j=1}^{N_S} \sigma(\vartheta_j^i, \varphi_j^i)$$

The inclusion of sign(M) ensures that minimising the aim function gives the appropriate result.

Example:

```
RCS
  1  3  90
```

Impedance / reflection factor

With this aim function, the input reflection factor of the antenna can be minimised. The optimisation can be broad band, by examining a number of frequencies at the same time. One can also optimise the reflection factor of multiple different ports simultaneously. The average reflection factor is minimised, with the option to limit values that have a sufficiently good match (to avoid that these are optimised further rather than focusing on the optimisation of other impedance values with a worse match).

The assignment of the aim function in the `*.opt` file is started by the word `IMPEDANZ` or `IMPEDANCE`. In the next row the parameters Z_{re}, Z_{im} and N are specified. Z_{re} is the real part and Z_{im} the imaginary part of the target impedance Z_{soll}. N is the number of blocks in the output file of FEKO, from which the impedance Z_i is read.

Example:

```
IMPEDANCE
  50  0  1
```

The aim function is defined as

$$Z = \frac{1}{N} \cdot \sum_{i=1}^{N} \left| \frac{Z_i - Z_{soll}}{Z_i + Z_{soll}} \right| \quad .$$

As a modification, one can specify an optional forth parameter in the line following the keyword `IMPEDANZ` or `IMPEDANCE`, for example:

```
IMPEDANCE
  50  0  1  -15.0
```

The fourth parameter is the minimum reflection coefficient $\Gamma_{\min}^{\mathrm{dB}}$ in dB, and if specified, the modified algorithm to compute the aim function Z is as follows.

For all impedance values $i = 1 \ldots N$ define the reflection coefficient

$$\Gamma_i = \left| \frac{Z_i - Z_{soll}}{Z_i + Z_{soll}} \right|$$

and then truncate to $\Gamma_{\min} = 10^{\frac{\Gamma_{\min}^{\mathrm{dB}}}{20}}$ (i.e. the corresponding linear desired reflection coefficient) in the summation:

$$Z = \frac{1}{N} \cdot \sum_{i=1}^{N} \max(\Gamma_i, \Gamma_{\min}) \ .$$

The reason for doing this is to avoid that one impedance value (such as one port or the impedance at one frequency) which has already an acceptable match is being optimised further and further, instead of focusing on the optimisation of other impedance values where the desired match has not yet been obtained.

Apart from the value of the aim function Z, the real part and the imaginary part of the input impedance Z_i are written to the log file (the latter only for $N = 1$).

Maximisation or minimisation of near fields

This aim function enables one to maximise the electric or magnetic field strength in the near field. This is particularly useful for a transmitting antenna that radiates constant power. In addition any linear combination — with arbitrary proportionality factors — of the electrical and magnetic fields can be maximised (or minimised by using negative values of the proportionality factors).

If only the electric or magnetic field must be maximised, the line following the keyword `NAHFELD` or `NEARFIELD` contains the parameters M, N_B and N_S. M is a flag and has the following meaning (depending on the coordinate system used with the near field

calculation in the selected block or blocks):

$$M = \begin{cases} 1 & \text{Use the } E_x;\ E_r;\ E_r \text{ components} \\ 2 & \text{Use the } E_y;\ E_\varphi;\ E_\vartheta \text{ components} \\ 3 & \text{Use the } E_z;\ E_z;\ E_\varphi \text{ components} \\ 4 & \text{Use the } H_x;\ H_r;\ H_r \text{ components} \\ 5 & \text{Use the } H_y;\ H_\varphi;\ H_\vartheta \text{ components} \\ 6 & \text{Use the } H_z;\ H_z;\ H_\varphi \text{ components} \end{cases}$$

The magnitude of the field strength is evaluated in each case. N_B gives the number of blocks, containing electric or magnetic near field strength values, to read and N_S gives the number of lines to read from each of these blocks.

For this case, the aim function Z is defined as

$$Z = -\frac{1}{N_B\, N_S} \cdot \sum_{i=1}^{N_B} \sum_{j=1}^{N_S} |E_{i,j}|$$

(using the magnetic near field $|H_{i,j}|$ when M is 4, 5 or 6). The negative sign ensures that a minimisation of the aim function maximises the near field.

Example:

```
NEARFIELD
  3  10  1
```

If a linear combination of the electric and magnetic fields must be maximised (it can be minimised by using negative factors) the keyword `NAHFELD` or `NEARFIELD` is followed by the parameters M, N_B^E, N_S^E, F^E, N_B^H, N_S^H and F^H in this order. The flag M is similar to the previous case

$$M = \begin{cases} 7 & \text{Use the } E_x/H_x;\ E_r/H_r;\ E_r/H_r \text{ components} \\ 8 & \text{Use the } E_y/H_y;\ E_\varphi/H_\varphi;\ E_\vartheta/H_\vartheta \text{ components} \\ 9 & \text{Use the } E_z/H_z;\ E_z/H_z;\ E_\varphi/H_\varphi \text{ components} \end{cases}$$

As before, only the magnitude of the field strength is evaluated in each case. N_B^E gives the number of electric near field blocks to read and N_S^E the number of lines to read from each of these blocks. Similarly N_B^H gives the number of magnetic near field blocks and N_S^H the number of lines for each block.

The parameters F^E and F^H are arbitrary proportionality constants that may also be negative. The aim function is now defined as

$$Z = -\frac{F^E}{N_B^E\, N_S^E} \cdot \sum_{i=1}^{N_B^E} \sum_{j=1}^{N_S^E} |E_{i,j}| - \frac{F^H}{N_B^H\, N_S^H} \cdot \sum_{i=1}^{N_B^H} \sum_{j=1}^{N_S^H} |H_{i,j}|$$

Example:

```
NEARFIELD
**    E-field          H-field
  7   10  1  1.0       10  1  376.73
```

Near field values

This aim function enables one to optimise either the electric or the magnetic near field or both with respect to certain defined field strength values. Since the absolute field strength is just scaling linear with the amplitude of the excitation, this extra degree of freedom is removed by supplying a normalisation of the near fields with respect to one point (whose amplitude after normalisation will then be one and then phase will be zero).

The optimisation parameters follow the keyword `NAHFELD_WERTE` or `NEARFIELD_VALUES`. If only the electric or the magnetic near-field is to be optimised to have a certain shape, then the line following this keyword contains the parameters:

$$M \;\; N_B \;\; N_S \;\; N_{norm} \;\; N_{comp} \;\; f_1 \;\; f_2 \;\; f_3$$

M is a flag and has the following meaning:

$$M = \begin{cases} 1 & \text{Use the electric near field} \\ 2 & \text{Use the magnetic near field} \end{cases}$$

N_B gives the number of blocks containing electric or magnetic near field strength values to be read, and N_S gives the number of lines to be read from each of these blocks. The value N_{norm} must be in the range $1 \ldots N_B\, N_S$ and indicates which field strength value shall be used for the normalisation.

Each field strength vector (either electric $\vec{E}$ or magnetic $\vec{H}$) has three components E_1, E_2, E_3 or H_1, H_2, H_3, respectively. For instance, in a Cartesian coordinate system the assignment is $E_1 = E_x$, $E_2 = E_y$ and $E_3 = E_z$, but in a spherical coordinate system one would have $E_1 = E_r$, $E_2 = E_\vartheta$, $E_3 = E_\varphi$. The value of N_{comp}, which must be 1, 2, or 3, determines which component of the N_{norm}th field strength value shall be used for the normalisation. The last three parameters f_1, f_2, and f_3 on the line following the keyword are weighting factors, which are discussed below when describing the aim function.

Let $E_{k,i,j}$ be the kth (with $k = 1, 2, 3$) component of the electric field strength in the ith block (with $i = 1 \ldots N_B$) and the jth line in this block (with $j = 1 \ldots N_S$), then we define normalised field strength values as

$$e_{k,i,j} = \frac{E_{k,i,j}}{E_{norm}}$$

where E_{norm} is the N_{comp}th component of the N_{norm}th field strength value that has been selected for the normalisation (this is a complex quantity with amplitude and phase). The aim function Z for the optimisation is then defined as

$$Z = \frac{1}{N_B\, N_S} \cdot \sum_{i=1}^{N_B} \sum_{j=1}^{N_S} \left[f_1 \left| e_{1,i,j} - e^{aim}_{1,i,j} \right|^2 + f_2 \left| e_{2,i,j} - e^{aim}_{2,i,j} \right|^2 + f_3 \left| e_{3,i,j} - e^{aim}_{3,i,j} \right|^2 \right] \quad .$$

The values $e^{aim}_{k,i,j}$ are the desired field values, and they must be specified in a tabular format in the `*.opt` file after the first two lines with the keywords and the parameters. There are in total $N_B\, N_S$ lines with all the field values, and each line has 6 columns with magnitude and phase (in degree) of the 3 normalised field strength components $e_{1,j,k}$, $e_{2,j,k}$ and $e_{3,j,k}$ (the inner loop is the one over j, i.e. the values appear in the same order as they would be in the output file):

$$|e_{1,i,j}| \quad \text{phase}(e_{1,i,j}) \quad |e_{2,i,j}| \quad \text{phase}(e_{2,i,j}) \quad |e_{3,i,j}| \quad \text{phase}(e_{3,i,j})$$

In the line N_{norm} of this table, the component N_{comp} (which we are normalising to) must be 1 in amplitude and 0 in phase.

So far only the case for the electric field has been discussed (normalisation, aim function, specification of values in the `*.opt` file), but the case of the magnetic field is analogous, one must just replace E by H and e by h in the formulas above.

Example:
Assume that one wants to optimise the electric near–field along a line to have a quadratic shape $(1 - 2\,\frac{x^2}{m^2})$ over 5 evenly spaced points in the range $-0.5\,\text{m} \le x \le 0.5\,\text{m}$, then the normalised field strength values are 0.5, 0.875, 1, 0.875, 0.5. We assume that only the z–component of the electric field is required to follow this shape, we don't care about the E_x and E_y components (but see comment below of how to enforce them to be zero for instance). Also the phase for all 5 points should be the same (then 0 because for the point no. 3 that we use for normalisation it has to be 0).

In the `*.pre file`, the FE card would look like

```
FE   1    5    1    1    0    -0.5      0         0         0.25
```

and in the `*.opt` file one would have the block

```
** Near field value optimisation, the first line
** after the keyword has the syntax
** M  NB  NS  Nnorm  Ncomp  fx   fy   fz
** and then follow NB*NS lines with the syntax
** |ex|  phase  |ey|  phase  |ez|  phase
NEARFIELD_VALUES
   1  1   5   3      3      0.0  0.0  1.0
   0.0   0.0    0.0   0.0    0.5   0.0
   0.0   0.0    0.0   0.0    0.875 0.0
   0.0   0.0    0.0   0.0    1.0   0.0
   0.0   0.0    0.0   0.0    0.875 0.0
   0.0   0.0    0.0   0.0    0.5   0.0
```

If one wants to enforce to have only E_z, but no E_x and E_y components, then all one has to do is to change the two factors f_1 and f_2 (which here in a Cartesian coordinate system refer to the x and y components) both to 1.0, so that these components are included in the aim function.

One can also optimise both electric and magnetic near fields simultaneously, then the syntax differs. The first line still has the keyword `NAHFELD_WERTE` or `NEARFIELD_VALUES`. After this we have a line

$$M \;\; N_B \;\; N_S \;\; N_{norm} \;\; N_{e,comp} \;\; N_{h,comp} \;\; f_{e,1} \;\; f_{e,2} \;\; f_{e,3} \;\; f_{h,1} \;\; f_{h,2} \;\; f_{h,3}$$

where the flag M must now be:

$$M = \Big\{ \; 3 \qquad \text{Use both the electric and magnetic near field}$$

The meaning of N_B, N_S and N_{norm} is still the same as when optimising only E or H, but for the component N_{comp} one can now specify this separately for the electric and magnetic field using $N_{e,comp}$ or $N_{h,comp}$, respectively. There appear now also 6 factors f in this line, they are used in the aim function

$$Z = \frac{1}{N_B\, N_S} \cdot \sum_{i=1}^{N_B} \sum_{j=1}^{N_S} \big[f_{e,1}\, |e_{1,i,j} - e^{aim}_{1,i,j}|^2 + f_{e,2}\, |e_{2,i,j} - e^{aim}_{2,i,j}|^2 + f_{e,3}\, |e_{3,i,j} - e^{aim}_{3,i,j}|^2 + f_{h,1}\, |h_{1,i,j} - h^{aim}_{1,i,j}|^2 + f_{h,2}\, |h_{2,i,j} - h^{aim}_{2,i,j}|^2 + f_{h,3}\, |h_{3,i,j} - h^{aim}_{3,i,j}|^2 \big] \quad .$$

The table with the required field strength values has now also both E and H field values in one line, the syntax is to have 12 columns

$$|e_{1,i,j}| \quad \text{phase}(e_{1,i,j}) \quad |e_{2,i,j}| \quad \text{phase}(e_{2,i,j}) \quad |e_{3,i,j}| \quad \text{phase}(e_{3,i,j}) \quad |h_{1,i,j}| \quad \text{phase}(h_{1,i,j}) \quad |h_{2,i,j}| \quad \text{phase}(h_{2,i,j}) \quad |h_{3,i,j}| \quad \text{phase}(h_{3,i,j})$$

(all in one line).

Of course this general case with $M = 3$ includes the cases that one optimises only for the electric or only for the magnetic field, one can just set all the factors $f_{e,\nu}$ to zero for instance and would then optimise only for the magnetic field. But the format of the table will then still require the specification of in this case arbitrary field values also for e, whereas when using $M = 2$ the format of the table of the required field strength values is simpler and includes only the desired normalised magnetic near field h.

10.4 Running OPTFEKO

Firstly the `*.pre` and `*.opt` files must be created as discussed above. During optimisation new `*.pre` input files are continuously created by adding the string `_opt_` and a sequentially incremented number to the file name. When, for example, the files `dipole.pre` and `dipole.opt` have been created, OPTFEKO is run with the command:

```
optfeko dipole
```

On the command line the following parameters can be added:

-r	The `*.pre` and `*.out` are deleted after each analysis. This saves disk space.
-R	Same as "-r", but the files of the optimum solution are kept.
-z	The value of the aim function is calculated for one existing file — no optimisation is done. This is mostly used for debugging.
-np x	If this option is given, the parallel version of FEKO (if it is available in the installation) is used in the solution. The parameter `x` specifies the required number of processes. This option is only supported on UNIX.

Information on the optimisation process is stored in a log file with the extension `.log` — in the example above the filename will be `dipole.log`.

10.5 An example using OPTFEKO

In figure 10-1 a dipole antenna in front of a reflector is shown. The gain is to be optimised. The bent angle α of the dipole and the distance a between the dipole and the reflector are given as optimisation parameters.

The input file `dipole.pre` for this geometry is as follows

```
** Optimisation Example
**
** A bent dipole antenna with variable distance in front of a
** square reflector
```

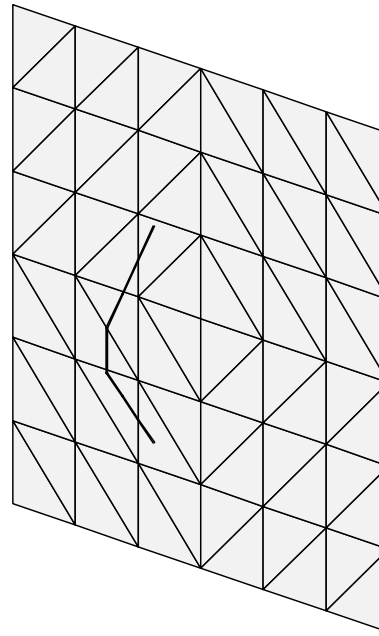


Figure 10-1: Bent dipole in front of a reflector

```
** The optimisation parameters are created by OPTFEKO
** Here we check if they have been defined (using the defined function) and
** define them such that we can run PREFEKO while working with this file.
!!if (not(defined(#a))) then
#a = 0.25
!!endif
!!if (not(defined(#alpha))) then
#alpha = 30
!!endif

** Remaining parameters
#lambda = 1
#seglen = #lambda/10
#segrad = #lambda/1000
#side_l = #lambda/5

** Set the segmentation parameters
IP                            #segrad   #side_l   #seglen

** Define points
```

```
DP     P1                      0.0       0.0       0.0
DP     P2                      0.0       #lambda/2 0.0
DP     P3                      0.0       #lambda/2 #lambda/2
DP     P4                      0.0       0.0       #lambda/2
#x = #a*#lambda - #lambda/4*sin(rad(#alpha))
#z = #lambda/4*cos(rad(#alpha))
DP     A                       #x        0.0       #z
#z = 0.45*#seglen
DP     B                       #a        0.0       #z
DP     C                       #a        0.0      -#z

** Define a quarter of the plate in the quadrant y>0 and z>0
BP     P1   P2   P3   P4
** Mirror the plate around the plane y=0 (xz-plane) -- ideal magnetic wall
SY   1    0    3    0

** Create the upper half of the dipole antenna (without feed segment)
** The default label (0) is still in use
BL     A    B
** Mirror in the plane z=0 (xy-plane) -- ideal electric conducting plane
SY   1    0    0    2

** Create the feed segment with the label 1
LA     1
BL     B    C

** End of the geometric input
EG   1    0    0    0    0

** Set the frequency, #c0 is the speed of light in vacuum
#freq = #c0/#lambda
FR   1                         #freq
** Excitation by means of a voltage gap (E field) at a node
A1   0    1                    1.0       0.0

** Calculate the far field
FF   1    1    1    0          90.0      0.0       0.0       0.0

** End
EN
```

As a first step, a discrete search is done by systematically varying the two optimisation parameters to find the region of the minimum.

The `*.opt` file is as follows

```
** Input file for the optimiser OPTFEKO
** for a bent dipole in front of a reflector

** Define optimisation parameters
** Name      Begin value     Minimum         Maximum
#alpha         -80            -80              80
#a            0.25          0.25             2.0

** Select the optimisation method (discrete search)
GRID_SEARCH
  20  20

** Select the aim function:
** (Gain, horizontal and vertical polarisation, 1 data block)
GAIN
   0   1

** End
```

The gain (the negative aim function) can be displayed graphically. In figure 10-2 one can see a clear maximum in the area $\alpha \approx 10^\circ$ and $\frac{a}{\lambda} \approx 0.8$.

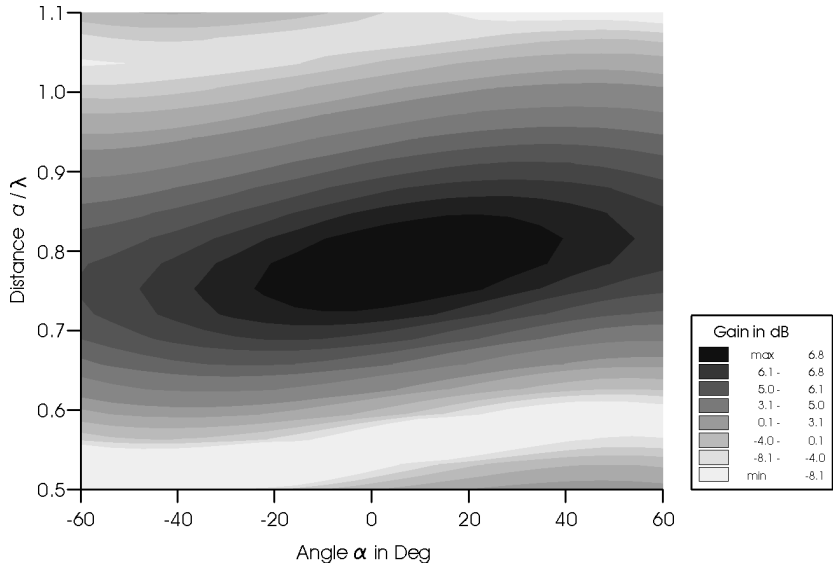


Figure 10-2: Gain as a function of the optimisation parameters

The general "position" of the optimum is now known. A new search can now begin with e.g. the simplex method, by using the input `*.opt` below

```
** Input file for the optimiser OPTFEKO
** For a bent dipole in front of a reflector, geometry in dipole.pre

** Define optimisation parameters
** Name         Begin value    Minimum        Maximum
#alpha         20             -20            40
#a             0.8            0.7            0.9

** Define penalty functions outside the optimisation region
PENALTY_FUNCTION
    100     100
     10      10

** Optimisation with the Simplex method
SIMPLEX_METHOD
  0.15    0.5  1.0E-4   1.0E-4

** Optimise the gain  (both polarisations; only 1 data block)
GAIN
   0   1

** End
```

The extract from the log file reproduced below shows that the procedure converges for $\alpha \approx 7^\circ$ and $\frac{a}{\lambda} \approx 0,78$

```
========================= Optimization and Analysis =========================

No.   #alpha      #a          Targetfct.  Penaltyfct.  other spec.
  1 2.0000e+001 8.0000e-001  -7.1538e+000 0.0000e+000 0.0000e+000   1 -7.1538e+000
  2 2.8693e+001 8.0776e-001  -7.0386e+000 0.0000e+000 0.0000e+000   1 -7.0386e+000
  3 2.2329e+001 8.2898e-001  -7.0007e+000 0.0000e+000 0.0000e+000   1 -7.0007e+000
  4 2.6364e+001 7.7879e-001  -7.0176e+000 0.0000e+000 0.0000e+000   1 -7.0176e+000
  5 1.7671e+001 7.7102e-001  -7.1202e+000 0.0000e+000 0.0000e+000   1 -7.1202e+000
  6 1.1307e+001 7.9224e-001  -7.2145e+000 0.0000e+000 0.0000e+000   1 -7.2145e+000
  7 1.3636e+001 8.2121e-001  -7.0740e+000 0.0000e+000 0.0000e+000   1 -7.0740e+000
  8 1.5653e+001 7.9612e-001  -7.1914e+000 0.0000e+000 0.0000e+000   1 -7.1914e+000
  9 1.2471e+001 8.0672e-001  -7.1658e+000 0.0000e+000 0.0000e+000   1 -7.1658e+000
 10 1.4489e+001 7.8163e-001  -7.1970e+000 0.0000e+000 0.0000e+000   1 -7.1970e+000
 11 1.0142e+001 7.7775e-001  -7.2171e+000 0.0000e+000 0.0000e+000   1 -7.2171e+000
 12 6.9600e+000 7.8835e-001  -7.2225e+000 0.0000e+000 0.0000e+000   1 -7.2225e+000
```

```
 13 5.7953e+000 7.7386e-001  -7.2218e+000 0.0000e+000 0.0000e+000   1 -7.2218e+000
 14 2.6133e+000 7.8447e-001  -7.2150e+000 0.0000e+000 0.0000e+000   1 -7.2150e+000
 15 3.7780e+000 7.9896e-001  -7.1728e+000 0.0000e+000 0.0000e+000   1 -7.1728e+000
 16 8.1247e+000 8.0284e-001  -7.1769e+000 0.0000e+000 0.0000e+000   1 -7.1769e+000
 17 9.1333e+000 7.9029e-001  -7.2205e+000 0.0000e+000 0.0000e+000   1 -7.2205e+000
 18 7.5423e+000 7.9560e-001  -7.2053e+000 0.0000e+000 0.0000e+000   1 -7.2053e+000
 19 8.5510e+000 7.8305e-001  -7.2270e+000 0.0000e+000 0.0000e+000   1 -7.2270e+000
 20 6.3777e+000 7.8111e-001  -7.2282e+000 0.0000e+000 0.0000e+000   1 -7.2282e+000
 21 7.9686e+000 7.7581e-001  -7.2214e+000 0.0000e+000 0.0000e+000   1 -7.2214e+000
 22 5.7953e+000 7.7386e-001  -7.2218e+000 0.0000e+000 0.0000e+000   1 -7.2218e+000
 23 4.2043e+000 7.7917e-001  -7.2255e+000 0.0000e+000 0.0000e+000   1 -7.2255e+000
 24 4.7867e+000 7.8641e-001  -7.2207e+000 0.0000e+000 0.0000e+000   1 -7.2207e+000
 25 7.4643e+000 7.8208e-001  -7.2281e+000 0.0000e+000 0.0000e+000   1 -7.2281e+000
 26 6.6688e+000 7.8473e-001  -7.2268e+000 0.0000e+000 0.0000e+000   1 -7.2268e+000
 27 7.1732e+000 7.7846e-001  -7.2266e+000 0.0000e+000 0.0000e+000   1 -7.2266e+000
 28 6.0865e+000 7.7749e-001  -7.2266e+000 0.0000e+000 0.0000e+000   1 -7.2266e+000
 29 5.2910e+000 7.8014e-001  -7.2274e+000 0.0000e+000 0.0000e+000   1 -7.2274e+000
 30 5.5822e+000 7.8376e-001  -7.2262e+000 0.0000e+000 0.0000e+000   1 -7.2262e+000
 31 6.9210e+000 7.8159e-001  -7.2283e+000 0.0000e+000 0.0000e+000   1 -7.2283e+000
 32 6.5232e+000 7.8292e-001  -7.2279e+000 0.0000e+000 0.0000e+000   1 -7.2279e+000
 33 6.7754e+000 7.7978e-001  -7.2279e+000 0.0000e+000 0.0000e+000   1 -7.2279e+000
 34 7.3187e+000 7.8027e-001  -7.2277e+000 0.0000e+000 0.0000e+000   1 -7.2277e+000
 35 7.4643e+000 7.8208e-001  -7.2281e+000 0.0000e+000 0.0000e+000   1 -7.2281e+000
 36 7.0666e+000 7.8341e-001  -7.2279e+000 0.0000e+000 0.0000e+000   1 -7.2279e+000
 37 7.1927e+000 7.8184e-001  -7.2282e+000 0.0000e+000 0.0000e+000   1 -7.2282e+000
 38 6.9938e+000 7.8250e-001  -7.2282e+000 0.0000e+000 0.0000e+000   1 -7.2282e+000

Optimisation finished (standard dev. small enough: 5.7735e-005)

Result destination vector: #alpha = 6.9209918e+000
                           #a = 7.8159398e-001

Minimal value of the aim function (at no. 31): -7.2283000e+000
no. of the last analysis: 38
```

11 The program TIMEFEKO

11.1 Description

With the program TIMEFEKO electromagnetic scattering problems can be solved in the time domain. It is based on the program FEKO, that does the calculation in the frequency domain, and an FFT algorithm, that transforms the data to the time domain.

For the excitation a number of different pulses have been defined and stored in the function library. The functions available at present are described in section 11.3.1.

The program TIMEFEKO is constructed in such a way that all the data in the output file `*.out` from FEKO are transformed, i.e. in the input file the cards have to be specified in the correct way to ensure that the correct data is transformed. Information on the correct card definitions is given in the following section.

The program TIMEFEKO uses two input files, that have to be created by the user.

11.2 The *.pre input file

The input file with extension `*.pre` is the normal input file for PREFEKO, in which the frequency *has* to be expressed symbolically. While the program TIMEFEKO is running, the frequency is constantly changed. TIMEFEKO generates new `*.pre` files in which the actual numerical value of the frequency is assigned. The user may use the DEFINED function to assign a numerical value to the frequency variable `#freq` in the `*.pre` — this is useful for running PREFEKO and viewing the geometry in WinFEKO (see section 11.5).

The FR card *must* use the frequency variable `#freq`. The number of frequencies to be examined *must* be one (NFREQ=1).

Example:

```
** Set the frequency
FR       1                              #freq
```

In the input file `*.pre` all the desired output parameters in their respective cards must be set (for example, FF card, FE card, OS card, ...). Only the data in the output file `*.out` can be transformed.

Note:

The program TIMEFEKO does not check the `*.pre` file. Special care should be taken to ensure that the segmentation parameters for the required frequency interval are fine enough for the program FEKO (see IP card, section 8.2.17). If they are not strictly adhered to, the program can terminate with an error message.

It is also possible to vary the segmentation for certain applications (far field, near field) to save memory and computation time. To do this one may use the variable `#freq` (which is constantly changed by TIMEFEKO) in the definition of the segmentation parameters.

Example:

```
** Define some constants
#maxfreq = 250.0e+08
#minlambda = #c0/#maxfreq

** Define the edge length  -  note the use of #freq
#edgelen = (2.0-#freq/#maxfreq)*#minlamdba/4.0

** Set the segmentation parameters
IP                                      #edgelen
```

11.3 The *.tim input file

In the file `*.tim` the pulse form, position and the point in time are assigned. This file consists of a number of sections which are optional at present.

- Set the pulse form with characteristic value and the time shift
- Set the size of the highest frequency and the sampling points
- Set the normalisation (normalise time to the speed of light)
- Set whether the output is written to the output file
- Set the time points

In the `*.tim` file, empty or comment lines starting with `**` are allowed. The parameters need not be entered in any particular column, but they have to be in the correct order.

The keywords used in the `*.tim` file exist in German and English (for example `ANREGUNG` and `EXCITATION`). Both versions of each keyword will be given in the discussion below. TIMEFEKO will recognise keywords in either language, independent of the language selected by the environment variable `FEKOLANG`.

11.3.1 Defining the pulse form

The assignment of the pulse form is necessary. Each pulse has a predefined name and has particular characteristic parameters. The parameters must be assigned absolute values and not normalised values. The amplitude factor u_0 is 1. It can be changed by using the Ax card (in the file `*.pre`) with the appropriate amplitude value.

Time shifting t_0: indicates the time (in seconds) that the pulses are delayed (see the shift in figure 11-1). The time shift should be such that the excitation of the structure 0 at $t = 0$ is approximately 0. Since the both the time and frequency domain data are continuous this is not strictly required, but it simplifies working with the results.

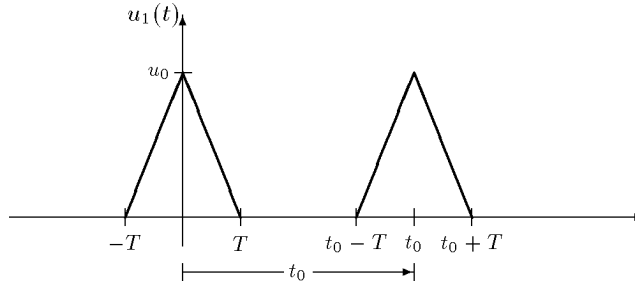


Figure 11-1: Time function $u_1(t)$ shifted by t_0

The following pulses are available:

1. Gaussian pulse `GAUSS / GAUSS` (see figure 11-2)

$$u_1(t) = u_0 \, \mathrm{e}^{-a^2(t-t_0)^2} \tag{11-1}$$

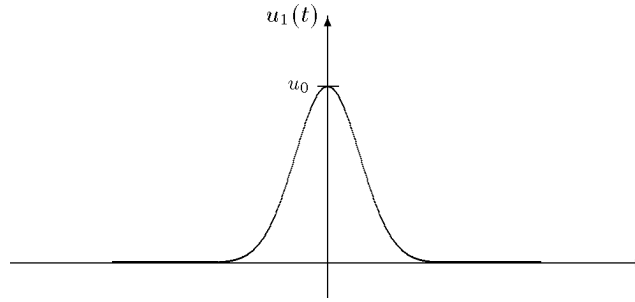


Figure 11-2: Gaussian pulse

Example:

```
** Pulse form
GAUSS
**      t0          Exponent a
    2.0E-08          3.0E+08
```

2. Triangular pulse `DREIECK / TRIANGLE` (see figure 11-3)

$$u_1(t) = \begin{cases} u_0 \left(1 - \dfrac{|t-t_0|}{T}\right) & \text{for} \quad |t-t_0| \leq T \\ 0 & \text{for} \quad \text{otherwise} \end{cases} \tag{11-2}$$

Example:

```
** Pulse form
TRIANGLE
**      t0          Impuls Duration T
    2.0E-08              1.0E-08
```

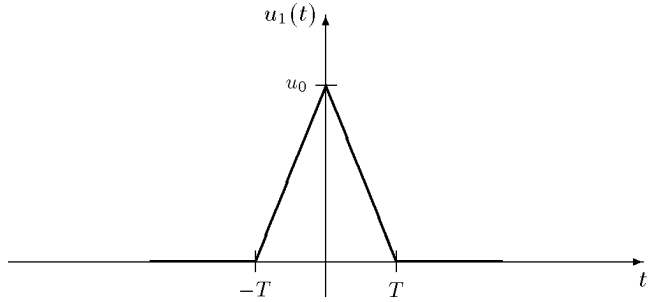


Figure 11-3: Triangular pulse

3. Double exponential pulse `DEXP / DEXP` (see figure 11-4)

$$u_1(t) = \begin{cases} 0 & \text{for} \quad t \le t_0 \\ u_1 \left(1 - \mathrm{e}^{-\frac{t-t_0}{\tau_1}}\right) & \text{for} \quad t_0 \le t \le T + t_0 \\ u_2 \, \mathrm{e}^{-\frac{t-t_0}{\tau_2}} & \text{for} \quad t \ge T + t_0 \end{cases} \tag{11-3}$$

$$u_1 = \frac{u_0}{1 - \mathrm{e}^{-\frac{T-t_0}{\tau_1}}} \qquad u_2 = \frac{u_0}{\mathrm{e}^{-\frac{T-t_0}{\tau_2}}} \tag{11-4}$$

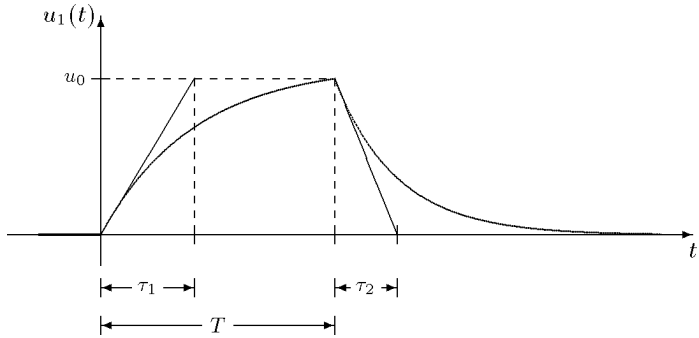


Figure 11-4: Double exponential pulse

Example:

```
** Pulse form
DEXP
**   t0          T                 tau1              tau2
     0.0         10.0E-09          5.0E-9            10.0E-9
```

4. Ramp pulse `RAMPE / RAMP` (see figure 11-5)

$$u_1(t) = \begin{cases} 0 & \text{for} \quad t \le t_1 + t_0 \\ u_0 \left(1 - \dfrac{|t - t_2 - t_0|}{\tau_1}\right) & \text{for} \quad t_1 + t_0 \le t \le t_2 + t_0 \\ u_0 & \text{for} \quad t_2 + t_0 \le t \le t_3 + t_0 \\ u_0 \left(1 - \dfrac{|t - t_3 - t_0|}{\tau_2}\right) & \text{for} \quad t_3 + t_0 \le t \le t_4 + t_0 \\ 0 & \text{for} \quad t \ge t_4 \end{cases} \tag{11-5}$$

$$t_1 = -\frac{T + \tau_1}{2} \qquad t_2 = -\frac{T - \tau_1}{2} \qquad t_3 = \frac{T - \tau_2}{2} \qquad t_4 = \frac{T + \tau_2}{2} \tag{11-6}$$

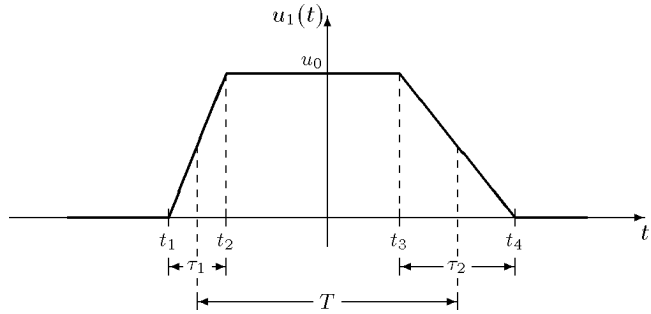


Figure 11-5: Ramp pulse

Example:

```
** Pulse form
RAMP
**   t0          Impuls Duration T      tau1              tau2
   20.0E-09         15.0E-09          5.0E-09           10.0E-09
```

5. Double exponential impulse (second type) `DBLEXP / DBLEXP` (see figure 11-6)

$$u_1(t) = \begin{cases} 0 & \text{for} \quad t \le t_0 \\ u_0 \left(\mathrm{e}^{-\frac{t-t_0}{\tau_1}} - \mathrm{e}^{-\frac{t-t_0}{\tau_2}}\right) & \text{for} \quad t > t_0 \end{cases} \tag{11-7}$$

Example:

```
** Impulse form
DBLEXP
** Time. t0      Parameter tau1     Parameter tau2
     20.0E-9         70.0E-9            5.0E-9
```

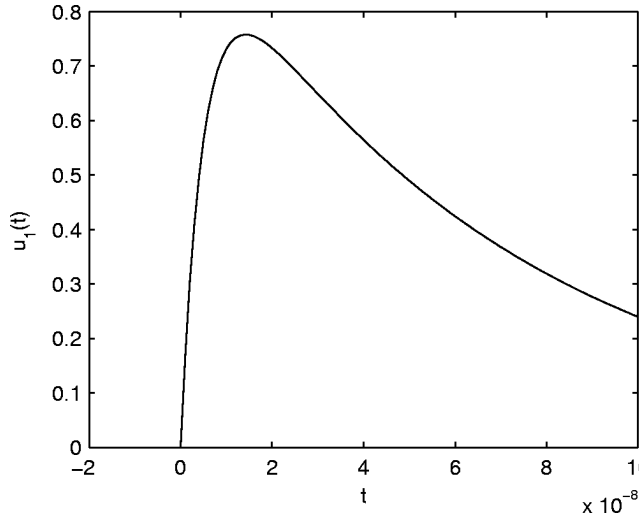


Figure 11-6: Double exponential impulse (second type)

11.3.2 Definition of the frequency block

The upper frequency limit f_{max} and the number of frequency points N are defined in the frequency block `FREQUENZ / FREQUENCY`.

Example:

```
FREQUENCY
** Upper frequency limit          Number of frequency points
          250.0E+06                          34
```

The maximum frequency f_{max} should be large enough such that the whole spectrum of the exciting pulse is covered. For example, for the Gaussian pulse `GAUSS` of section 11.3.1 we find

$$f_{3dB} = \frac{a}{\pi}\sqrt{\ln\sqrt{2}} \approx 0.187\, a$$

with a as defined in equation 11-1. One should select approximately $f_{max} = 4\, f_{3dB}$.

The number of frequency points N is then selected such that the total time T is long enough for the exciting pulse and all included currents, radiated fields, etc. to have decayed. Once one has determined the total time, the number of samples may be determined from the relation

$$\Delta f = \frac{1}{T} = \frac{f_{max}}{(N-1)}$$

or

$$N = 1 + T\, f_{max} \tag{11-8}$$

Finally, let P be the smallest power of 2 which is larger than or equal to $N-1$. (For example: for N=10, P=16; for N=33, P=32; and for N=50, P=64.) Then the time stepping will be $\Delta t = \frac{T}{2P} = \frac{N-1}{2\,f_{max}\,P}$. With the relation $\frac{P}{2} < N-1 \leq P$ we get the bounds

$$\frac{1}{4\,f_{max}} < \Delta t \leq \frac{1}{2\,f_{max}}$$

11.3.3 Definitions of the normalisation

Using the keyword `NORM / NORM`, the time can be normalised to the speed of light in a vacuum c_0. The normalised time then has a unit of lm *(light-metre)*.

$$t_{norm} = t \cdot c_0$$

Example:

```
** Normalising time with respect to the speed of light
NORM
```

11.3.4 Definition of the excitation output

The keyword `ANREGUNG / EXCITATION` indicates whether the time variation of the exciting pulse is to appear in the output file.

Example:

```
** Output the excitation pulse
EXCITATION
```

11.3.5 Definition of a time point

With the keyword `ZEITPUNKTE / POINTS_IN_TIME`, the near fields, surface currents and line currents can be calculated at certain points in time. Using time points has no effect on the calculation of source currents or far fields. If normalisation has been set (`NORM`), then the time points must also be given in normalised form.

Example:

```
** Specifying the time points -- time has been normalised
POINTS_IN_TIME
  5.0   10.0  15.0  20.0
```

11.4 Running TIMEFEKO

Firstly, the input files `*.pre` and `*.tim` have to be created. During execution new input files `*.pre` are continuously generated by adding the string `_tim_` and a sequentially incremented number to the file name. If, for example, the input files `cube.pre` and `cube.tim` have been created, TIMEFEKO is executed with the command

```
timefeko cube
```

On the command line the following parameters can be added:

-a	Here an FFT is only performed on the already available data
-r	The `*.pre` and `*.out` files are deleted after each iteration. This saves disk space.
-np x	If this option is given, the parallel version of FEKO (if it is available in the installation) is used in the solution. The parameter x specifies the required number of processes. This option is only supported on UNIX.

11.4.1 TIMEFEKO output

TIMEFEKO generates different `*.pre` files at the different frequencies. The `*.out` files for all these runs are available if TIMEFEKO was called without the `-r` option. These results are summarised — if they are requested in the `*.pre` file — in `*.ost` (triangle currents), `*.oss` (segment currents), `*.cur` (the currents on voltage sources), `*.far` (far fields), `*.nfe` (near electric fields) and `*.nfh` (near magnetic fields). In each case the results for each frequency is listed one after the other with frequencies separated by a line containing a single #-character.

The time domain results are given in the `*.aus` file for all outputs requested in the `*.pre` file. Currents are requested by the OS card. Note that requesting the current on a large number of structures will result in very large `*.aus` files.

11.5 A TIMEFEKO example

In figure 11-7 an ideal conducting metallic cube with side lengths of 1m is shown. The current in the middle of the front side, the scattered field from the direction of incidence (the incident wave travels in the negative x direction), as well as the excitation pulse are to be calculated. The input file `cube.pre` is reproduced below. (These files are located in the `examples\utils\timefeko` subdirectory of the FEKO installation.)

```
** TIMEFEKO example (*.pre file)
** A metallic cube with side lengths 1m. Only 1/8 of the cube is generated
** explicitly, the rest of the cube is generated by means of symmetry.
```

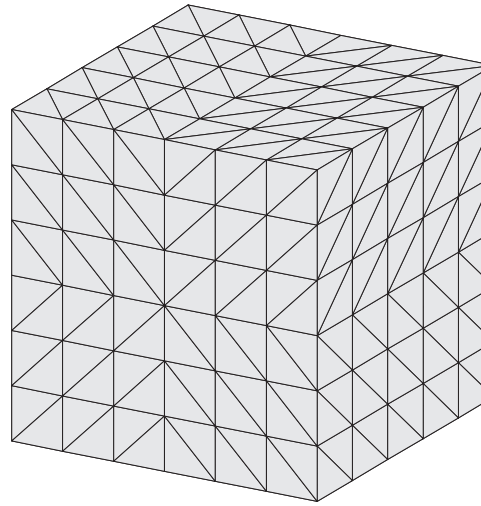


Figure 11-7: Cube with side lengths of 1m

```
** Normally TIMEFEKO will automatically insert the correct required
** frequency value. Use the following construct so that this value
** used by TIMEFEKO will not be overwritten, but we can still display
** the geometry in WinFEKO.
!!if not defined(#freq) then
#freq = 100.0e6
!!endif

** Define some constants
#a = 1           ** side length of the cube
#edgelen = #a/5  ** max. edge length for the triangular patches

** Set the segmentation parameters
IP                                      #edgelen

** Define the points
DP     P1                       #a/2      0         0
DP     P2                       #a/2      #a/2      0
DP     P3                       #a/2      #a/2      #a/2
DP     P4                       #a/2      0         #a/2
DP     P5                       0         0         #a/2
DP     P6                       0         #a/2      #a/2
DP     P7                       0         #a/2      0
```

```
** Create one eigth of the cube (use label 1 for the front plate and
** label 0 for the rest)
LA   1
BP     P1   P2   P3   P4
LA   0
BP     P3   P4   P5   P6
BP     P2   P3   P6   P7

** Mirror around to coordinate planes so that label for front plate
** remains 1, all other surfaces will have label 0)
**   x=0 (yz-plane): only geometric symmetry
**   y=0 (xz-plane): ideal magnetic conducting plane
**   z=0 (xy-plane): ideal electric conducting plane
SY   1    1    0    0    1
CB   2    0
SY   1    0    3    2

** End of the geometry
EG   1    0    1    0    0

** Set the frequency
FR   1    0                   #freq

** Excitation by means of an incident plane wave
AO   0         1    1          1.0       0.0       90.0      0.0       0.0

** Surface current density output for surface with label 1
OS   4    1    1

** Calculate the far field only in the direction of incidence
FF   2

** End
EN
```

For this example we have chosen a Gaussian pulse excitation with $a = 3 \times 10^8$. As discussed in section 11.3.2, f_{3dB} is approximately 56 MHz such that we require at a maximum frequency of at least 224 MHz. We select a maximum frequency of 250 MHz. The time shift selected for this example is 6 light-metre and the structure dimensions is of the order of 1 metre. Thus we believe that the time response should die out within 40 light-metre or 133 ns. Then equation 11-8 then yields $N = 34$.

The `*.tim` input file `cube.tim` then contains

```
** Timefeko Example (*.tim file)

** Define the Pulse form
GAUSS
** Parameters of the Gaussian pulse
**   Time shift      Exponent
        2.0e-8          3.0e+8

** Define the frequency block
**   Gaussian pulse with a=3.0e+8 1/s, i.e. f_3dB = 0.187*a = 56.2 MHz
**   Choose f_max > 4*f_3dB = 224.9 MHz, use f_max = 250 MHz
**   Total time we want to analyse T = 40 lightmetres = 133.4 ns,
**   i.e. N=1+T*f_max = 34
FREQUENCY
** Upper frequency     Number of Samples
        250.0e+06                34

** Normalise the time to that of the speed of light
NORM

** Output the excitation
EXCITATION
```

The following is an extract from the output file, `cube.aus`

```
          TEMPORAL VARIATION OF EXCITATION NORMALISED TO U_0

    x       y       z
   0.0     0.0     0.0
Time in lm       Value
0.0000000e+000  2.31952283024e-016
3.0916097e-001  8.63275340216e-015
6.1832194e-001  2.65316865993e-013
9.2748292e-001  6.73356755347e-012
1.2366439e+000  1.41120603102e-010
1.5458049e+000  2.44230818918e-009
1.8549658e+000  3.49040118194e-008
2.1641268e+000  4.11922076322e-007
2.4732878e+000  4.01439061167e-006
2.7824487e+000  3.23064343845e-005

....
```

```
      VALUES OF THE SCATTERED ELECTRIC FIELD STRENGTH IN THE FAR FIELD in V
                    Factor e^(-j*BETA*R)/R not considered

   THETA     PHI
  90.00     0.00
  Time in lm          ETHETA               EPHI
0.0000000e+000 -2.98948137392e-005  0.00000000000e+000
3.0916097e-001 -6.77873335003e-005  0.00000000000e+000
6.1832194e-001  4.51882462221e-005  0.00000000000e+000
9.2748292e-001  6.42849147499e-005  0.00000000000e+000
1.2366439e+000 -6.07906709346e-005  0.00000000000e+000
1.5458049e+000 -4.10609464655e-005  0.00000000000e+000
1.8549658e+000  1.46284430530e-004  0.00000000000e+000
2.1641268e+000  3.56160952549e-004  0.00000000000e+000
2.4732878e+000  1.37768474311e-003  0.00000000000e+000
2.7824487e+000  5.67653919004e-003  0.00000000000e+000
3.0916097e+000  1.78441744890e-002  0.00000000000e+000
3.4007707e+000  4.45400653850e-002  0.00000000000e+000
3.7099317e+000  8.92614320231e-002  0.00000000000e+000
4.0190926e+000  1.40179442097e-001  0.00000000000e+000

...


           VALUES OF THE CURRENT DENSITY VECTOR ON TRIANGLES in A/m (no averaging)
number            x/m          y/m          z/m
       1   5.00000E-01  5.55556E-02  5.55556E-02
  Time in lm          JX                JY                JZ
0.0000000e+000  0.00000000000e+000  3.80396548298e-009 -2.65565071677e-007
3.0916097e-001  0.00000000000e+000 -3.29242237538e-009  1.93726675808e-006
6.1832194e-001  0.00000000000e+000 -2.60940425868e-009  1.37480675066e-006
9.2748292e-001  0.00000000000e+000  4.56971413492e-009 -9.30586425956e-007
1.2366439e+000  0.00000000000e+000  2.25486652232e-009 -7.87315697551e-007
1.5458049e+000  0.00000000000e+000 -4.83790802156e-009  1.35885783156e-006
1.8549658e+000  0.00000000000e+000 -2.80525051677e-010  1.63413481451e-006
...
number            x/m          y/m          z/m
       2   5.00000E-01  1.11111E-01  1.11111E-01
  Time in lm          JX                JY                JZ
0.0000000e+000  0.00000000000e+000 -1.18216923283e-008 -1.93459474753e-007
3.0916097e-001  0.00000000000e+000 -9.02561817670e-010  1.95358888266e-006
6.1832194e-001  0.00000000000e+000  1.28360949904e-008  1.34406393148e-006
9.2748292e-001  0.00000000000e+000  4.46200189557e-009 -9.71500429087e-007
1.2366439e+000  0.00000000000e+000 -4.91955007365e-009 -8.71581870796e-007
1.5458049e+000  0.00000000000e+000  9.27286771090e-010  1.21262288818e-006
1.8549658e+000  0.00000000000e+000  3.59541937732e-009  1.49801053069e-006
...
```

```
number            x/m           y/m           z/m
       3   5.00000E-01   5.55556E-02   2.22222E-01
  Time in lm          JX                JY                JZ
0.0000000e+000  0.00000000000e+000  9.87039561714e-009 -1.55875015522e-007
3.0916097e-001  0.00000000000e+000 -1.10165720719e-008  1.96750870513e-006
6.1832194e-001  0.00000000000e+000 -3.92248537619e-009  1.37781030380e-006
9.2748292e-001  0.00000000000e+000  1.95306492829e-008 -8.84072619127e-007
1.2366439e+000  0.00000000000e+000  6.20307013697e-009 -7.89929184543e-007
1.5458049e+000  0.00000000000e+000 -1.97321161834e-008  1.24837047986e-006
1.8549658e+000  0.00000000000e+000 -1.39083930925e-009  1.53916294269e-006

...
```

Figure 11-8 shows the response of the excitation $\vec{E}_i(t)$ in the time domain and figure 11-9 the back scattered electric far field E_z.

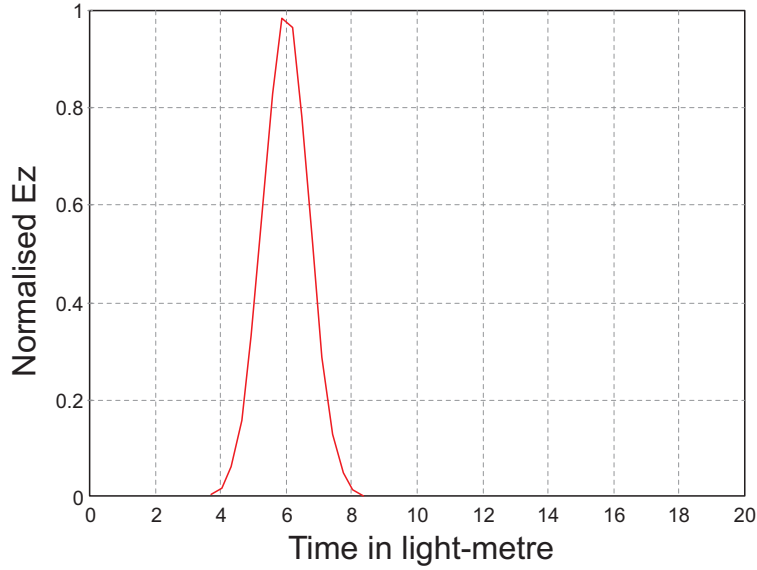


Figure 11-8: Time response of the excitation E_{iz}. The parameters of the Gaussian pulse are: $a = 3.0 \cdot 10^8\,\mathrm{s}^{-1}$, $t_0 = 20\,\mathrm{ns}$.

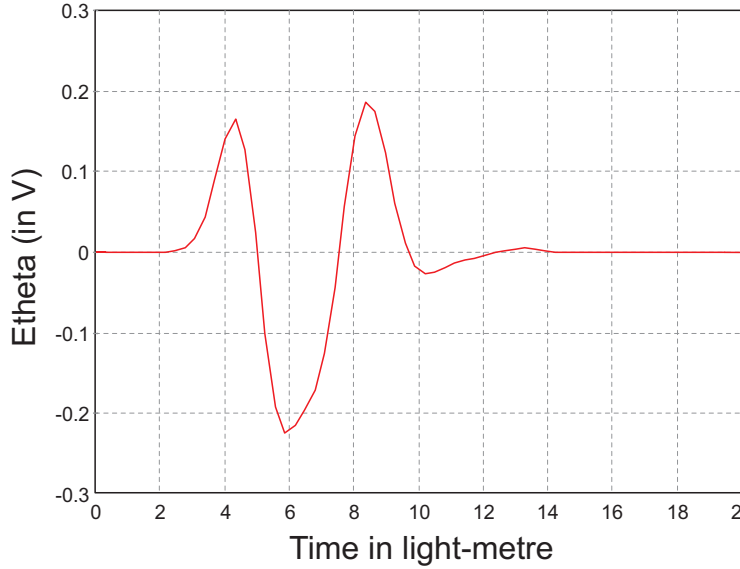


Figure 11-9: Response of the back scattered far field E_z of the cube.

12 The program LFFEKO

12.1 Description

The acronym LFFEKO is derived from *low frequency FEKO*. LFFEKO is not a special program, but rather a special execution mode of FEKO, where special basis functions for metallic surfaces are used to reduce the loss of numerical accuracy due to a finite number of decimal digits at low frequencies. These special basis functions are known as loop and star basis functions.

FEKO tests for numerical accuracy and responds with warning 594. If the standard FEKO version does not give warning 594, the use of LFFEKO is normally not required. LFFEKO is activated by means of the EG card in the `*.pre` input file (see description of the EG card in section 8.2.12).

Only metallic objects may be present in the input file, the use of wires and dielectric bodies is not possible. The use of symmetry is likewise also not possible in the current version. PO and UTD are also not allowed. Since wires are not allowed, excitations such as A1, A2 and A3 cannot be used.

In addition to the `*.pre` file, LFFEKO requires a `*.geo` file.

12.2 The *.pre input file

The input file with the extension `*.pre` is a normal input file for PREFEKO, except that in the EG card (see description of the EG card in section 8.2.12) the parameter LFFEKO has to be activated. See the above description of the limitations.

12.3 The *.geo input file

The input file with the extension `*.geo` serves the purpose of manually changing the loop order as well as to take sections of the geometrical structure. For the correct solution of the scattering problem, the number of unknown variables in the matrix equation has to be corrected if closed or connected surfaces are present.

Description of the File Format

The filename of the `*.geo` file has to be present in the first line, then the commands (see below) followed by `END` at the end. Comments may be placed between the commands.

The respective commands must be placed in the first column and have to be in capital letters. The command parameters are integers. For each line a maximum of ten arguments are allowed. Care needs to be taken to ensure that the arguments appear in the correct columns, which are 11–15 (first argument), 16–20 (second argument), 21–25 (third argument) etc.

12.3.1 CUTEDGE command

With this command the cuts can be added. An inner edge is separated by the indicated two adjacent triangles. After the command a pair of triangles is always expected, which have a common inner edge. The triangle with the second number is duplicated. The indicated inner edge of the second triangle becomes a free edge. The input method allows the transformation of a single basis function.

If the edge has more than one original basis function, as is the case for connected plates, a specific original basis function can be transformed.

If more than one edge is to be separated, then the entries follow in the next line. The input is completed with the command CUTEND.

Example:

```
** Cuts are added between the triangles
** 2 and 3 as well as between 8 and 19
CUTEDGE       2    3
              8   19
CUTEND
```

12.3.2 DELLOOP command

The arguments of the DELLOOP command are loop numbers. The loops entered by mean of a number are deleted. Per command a maximum of 10 loops can be deleted.

Example:

```
** Delete the loops 1, 5, 10 and 18
DELLOOP       1    5   10   18
```

12.3.3 INSLOOP command

Using INSLOOP a loop (super loop) can be created, as is required for apertures. The arguments are triangle numbers, that are to form the loop. If more than ten arguments are required, a second line can be used without using the command. The entry is ended with the command INSEND.

Example:

```
** Form a superloop over the triangles 1-15
INSLOOP       1    2    3    4    5    6    7    8    9   10
             11   12   13   14   15
INSEND
```

12.3.4 END command

This command is a necessity and indicates the end of the `*.geo` file.

12.4 Example

The surface current distribution on the surface of a perfect conduction sphere is required. The file `sphere.pre` describe the geometry which is shown (with the solution of the current) in figure 12-1. The file is as follows

```
** Sphere with 1m radius at 50 Hz
** Excited with a plane wave incident from the z direction
** and the H field polarised in the y direction

** Parameters for segmentation
IP                                      0.5
** Define points
DP     P1                      0.0       0.0       0.0
DP     P2                      0.0       0.0       1.0
DP     P3                      1.0       0.0       0.0
** Sphere with centre at (0,0,0), radius 1m
KU     P1   P2   P3   0        0.0       0.0       180.0     360.0     0.4

** End of the geometric input, specify low frequency FEKO version
EG   1    0    0    1    0
PS        0    0    1    0
** Frequency
FR        0                   50.0
** plane wave, so that Hi = 1 A/m
AO   0         1    1          #zf0      0.0       180.0     0.0       0.0
** Output the surface currents
OS   1         0

** End
EN
```

Note the 1 in the fourth integer field of the EG card. This activates LFFEKO which then requires an additional input file, `sphere.geo` reproduced below

```
sphere

** The sphere is a closed solid.  For these structures the number
** of loops must be reduced, otherwise the matrix will be singular

DELLOOP   1

END
```

The following is an extract from the output file `sphere.out`

```
              Special formulation for low frequencies
 1 loop(s) has(have) been deleted

...
              Condition number of the matrix:  3.85214E+03
...

           VALUES OF THE CURRENT DENSITY VECTOR ON TRIANGLES in A/m

Triangle    centre                      JX               JY               JZ
number    x       y       z      magn.    phase    magn.    phase    magn.    phase
    1     .949    .128     .128 2.823E-01  180.00 3.315E-05 -179.67 1.419E+00     .00 ...
    2     .949    .128    -.128 2.823E-01     .00 3.359E-05    -.33 1.419E+00     .00
    3     .897    .253    -.255 3.001E-01     .00 8.287E-03     .00 1.420E+00     .00
    4     .897    .253     .255 3.001E-01  180.00 8.287E-03 -180.00 1.420E+00     .00
    5     .825    .125    -.491 8.114E-01     .00 1.138E-06   -8.40 1.214E+00     .00
    6     .825    .125     .491 8.114E-01  180.00 6.077E-07 -164.10 1.214E+00     .00

...
```

The surface current on the surface of the sphere is shown in the figure 12-1. It should be noted that this example can also be solved in the normal version of FEKO, where symmetry can then be used. Using LFFEKO the conditioning number is reduced by approximately a factor of 10^{10}.

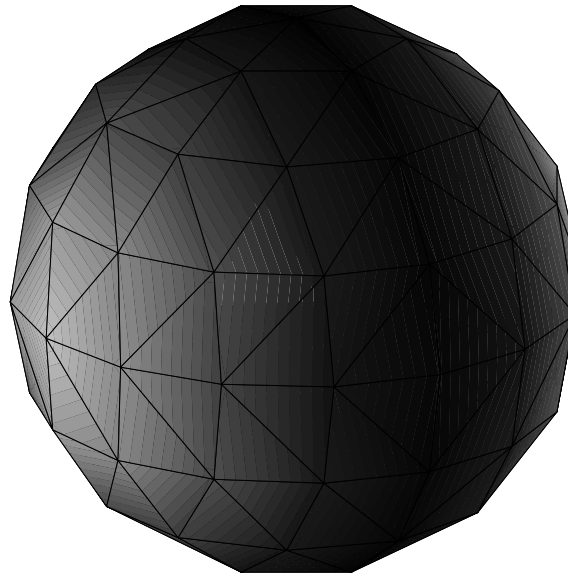


Figure 12-1: Surface Current Distribution on the surface of the sphere

13 The program ADAPTFEKO

13.1 Description

In examples with narrow resonances a fine frequency resolution is required to locate these resonances. If the frequency band is large, a very large number of analysis may be required if simple linear or multiplicative frequency stepping is used. ADAPTFEKO is used to overcome these problems. It uses an adaptive frequency sampling and interpolation, automatically using smaller steps near resonances and larger steps where the results are relatively smooth.

For each frequency it creates a `*.pre` file and calls PREFEKO and FEKO. The filenames are derived from the original name plus `_ada_` plus a numerical value (for example, the new files associated with `forked_dipole.pre` are `forked_dipole_ada_1.pre`, `forked_dipole_ada_2.pre`, ...).

13.2 Running ADAPTFEKO

ADAPTFEKO is started automatically by RUNFEKO if the FR card contains the flag for adaptive frequency sampling (see sections 7.2 and 9.2.26). The syntax is

```
runfeko filename
runfeko filename --adaptfeko-options options
```

where the optional argument `options` in the second line may be

`--keep-files`	All solution files (`*.pre`, `*.fek`, `*,out`, etc.) are preserved.
`--restart x`	Restart an adaptive frequency analysis using results for the frequency points `1...(x-1)` obtained in a previous run. (Then the previous run must have used `--keepfiles`.)

13.3 The *.pre input file

The `*.pre` file is created as for linear or multiplicative stepping. The only requirement is that only one FR card is used and that this card requests a continuous frequency band.

The variable `#adaptfreq` is defined automatically at the start of the single frequency input files and this variable may be used to allow, for example, adaptive meshing. One should not directly assign this variable inside the `*.pre` file as this will overwrite the value specified by ADAPTFEKO at the top of the file. If one needs this variable (for example to run PREFEKO during model setup when using adaptive meshing), one may use the DEFINED function

```
!!if (not(defined(#adaptfreq))) then
#adaptfreq = 250.0E6
!!endif
```

13.4 ADAPTFEKO example

As an example we will consider the input impedance of a simple forked dipole shown in figure 13-1. The input file `forked_dipole.pre` is as follows (this file is located in the `examples\utils\adaptfeko` subdirectory of the FEKO installation.)

```
** Forked dipole antenna
** Full MoM solution with adaptive frequency sampling

** Frequency (for the discretisation)
#lam = #c0 / 3.0e8

** Segmentation parameters
IP                              #lam/1000           #lam/20

** Define some points
DP   P1                         -0.01      0.0       0.5
DP   P2                          0.0       0.0       0.01
DP   P3                          0.01      0.0       0.466
DP   P4                          0.0       0.0      -0.01

** Half of the wire
LA   0
BL   P1   P2
BL   P2   P3

** Symmetry
SY   1    0    0    1

** The feed segment
LA   1
BL   P2   P4

** End of geometry
EG   1    0    0    0    0


** Excitation
A1   0    1                    1         0

** Adaptive frequency band
FR   100  2                    100.0e6   0.1e6     300.0e6
```

```
** Just compute the impedance
OS   0

** End
EN
```



Figure 13-1: Forked dipole used in the ADAPTFEKO example

Note that we do not use adaptive meshing as the model is quite small. This avoids the trouble associated with small discontinuities resulting from changes in the mesh.

After running ADAPTFEKO we have the file `forked_dipole.out` which is a combination of the output files at the sample frequencies. In addition, we have the file `forked_dipole.afo` which contains the continuous frequency data in a format which can be used by GraphFEKO. (The file `forked_dipole.bof` is a binary output file which will be used in a future release of FEKO.)

If the solution is done on a workstation both the `*.out` and `*.afo` files must be copied to the PC where the post processing is done. To display the results in GraphFEKO import the `*.out` file. If the `*.afo` file is available in the same directory, the *ADAPTFEKO results* button and the *Import* → *ADAPTFEKO results* menu item become enabled. Selecting either of these opens the *ADAPTFEKO results* panel in GraphFEKO. All parameters for which continuous data is available are listed under *Parameter* at the top. The other selections depend on the selected parameter.

For this example we select `Z (sol=1,source=1)` which refers to the input impedance of the first source encountered in the first solution block. (The solution counter is incremented each time a new solution is encountered, for example when adding a skin effect

and requesting a new solution.) In this example there is only one source and one solution. The remaining parameters are selected as required before creating the graph. The *Modify* button may be used to change the last set of lines added to the graph as long as the active graph has not changed. This is mostly used to change the number of frequency increments. (Note that the results are continuous, this selection specifies only the number of sample points used for the graphical representation.)

The resulting input impedance is given in figure 13-2. Note the grey marks at the bottom which give the sample frequencies used by ADAPTFEKO. (These marks can be switched of from the *Bottom axis* panel which may be accessed from the *Main graph settings* panel.) Figure 13-3 shows results over a small frequency band where a solution obtained with linear stepping is added to the same graph. Note how close the results match even though ADAPTFEKO used only a single sample in this region.

This example was taken from the paper "Efficient wide–band evaluation of mobile communications antennas using [Z] or [Y] matrix interpolation with the method of moments," by K. L. Virga and Y. Rahmat-Samii, in the *IEEE Transactions on Antennas and Propagation*, vol. 47, pp. 65–76, January 1999. In that paper they considered the input admittance of a forked monopole. For comparison we have plotted the input admittance, scaled it to mSiemens and scaled the bottom axis to h/λ with $h = 1$m. Since we consider a dipole while their example uses a monopole we need to multiply the admittance with a further factor of 2. The result is shown in figure 13-4 and compares very well to the published result.

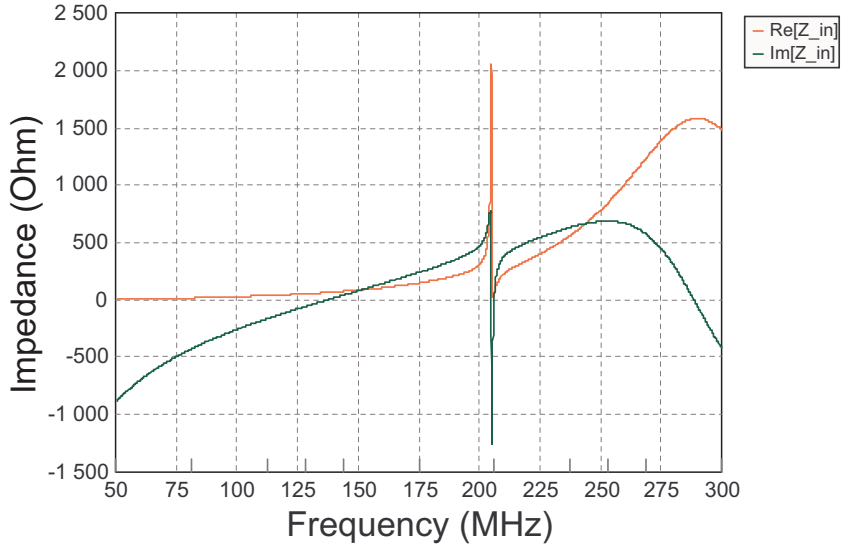


Figure 13-2: Input impedance of the forked dipole

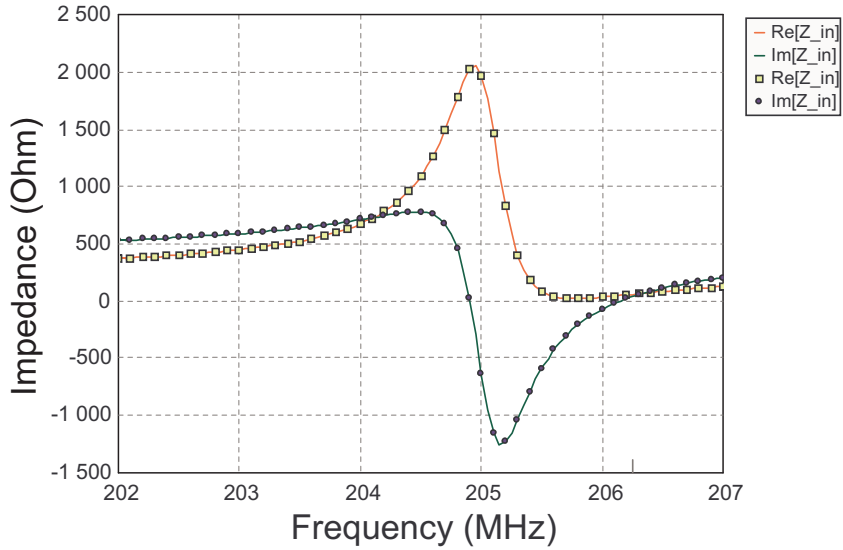


Figure 13-3: Input impedance of the forked dipole around the resonance point. The squares and circles represent values calculated at the discrete frequencies. The marker at 206.2 MHz denotes the only adaptive sample point in this band.

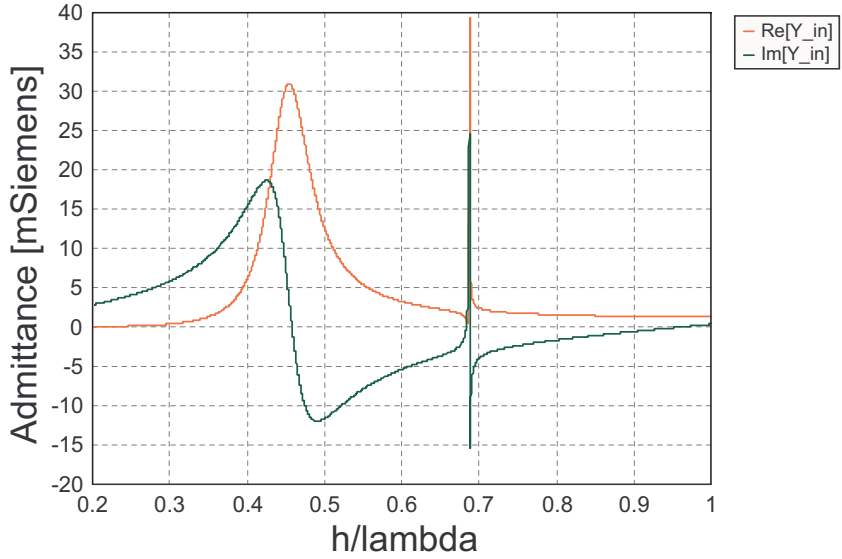


Figure 13-4: Input admittance of a forked monopole derived by multiplying the admittance of a forked dipole by a factor of 2.

14 Description of the output file of FEKO

The program FEKO writes all the results to an output file `*.out`. In this section the parts of the output file are described.

14.1 Geometric data

First the geometric data is given (if it has not been suppressed by the PS1 parameter in the EG card). For the metallic triangles the following extract is written:

```
                    DATA OF THE METALLIC TRIANGLES

 no. Label      x1 in m      y1 in m      z1 in m       edges
     medium     x2 in m      y2 in m      z2 in m
                x3 in m      y3 in m      z3 in m       area in m*m
   1    0        1.2733        .0000        .0000        1      2
        0        1.9100        .0000        .0000
                 1.7646        .7309        .0000      2.3268E-01
   2    0        1.1027        .6367        .0000       -1      3      4
        0        1.2733        .0000        .0000
                 1.7646        .7309        .0000      2.1874E-01
```

In the first column the number of the triangle is written. In the second column the label is given, followed by the medium in which the triangle is situated. A 0 means that it is in free space. The next three columns are the x, y and z coordinates of the three corner points of the triangles. In the first row of each triangle follows another list of the numbers of the edges of the adjacent triangles. A positive sign indicates that the positive current direction is away from the triangle. A negative sign indicates that the positive current direction is towards the triangle. Below the edge numbers the area of the triangle is given in m^2.

Following this is an extract of the data for the edges between the triangle. Whenever two triangles have two common vertices, such an edge is generated.

```
                    DATA OF THE METALLIC EDGES (with MoM)

                                                 triangle no.    points of tr. ...
  no.  type       length/m        media      KORP   KORM   POIP   POIM
     1   1    5.3033E-01       0     -1       1      2      1      1
     2   1    3.7500E-01       0     -1       1     33      2      3
     3   1    3.7500E-01       0     -1       1     65      3      2
                                                  information on symmetry
                                                yz     xz     xy  status
                                                 0     49     93  unknown
                                                 0     -2     94        0
                                                 0     50     -3  unknown
```

Each edge is assigned a consecutive number, which appears in the first column. The length of the edge is given in the second column and the medium in which the edge is found appears in the third column. On an edge there are exactly two triangles. In the columns KORP and KORM the numbers of these two triangles are given and the positive current direction is from the triangle KORP to the triangle KORM. In the column POIP the number of the corner point of the triangle KORP, which is opposite to the edge, is given. The same applies to the column POIM.

The next four columns contain information concerning the symmetry. In the column yz the number of the edge appears, corresponding to the plane $x = 0$ (yz plane) of symmetry. A positive sign indicates that the currents are symmetric and a negative sign indicates that the currents are anti-symmetric. If there is a 0 present in this column, then a symmetric edge does not exist. The same applies to the next columns xz and xy concerning the planes $y = 0$ and $z = 0$.

The last column with the heading STATUS has the following meaning: If unknown appears in it, the edge has an unknown status. The applicable coefficient of the current basis function cannot be determined from the symmetry, but has to be determined form the solution of the matrix equation. If 0 is present in the STATUS column, then the coefficient of the current basis function is 0 due to electric or magnetic symmetry and does not have to be determined.

If there is any other number in the STATUS column then this number indicates another edge whose coefficient is equal to (positive sign in the column STATUS) or the negative of (negative sign in the column STATUS) of the coefficient of the current basis functions. From symmetry the coefficient of the current triangle does not have to be determined.

The data of the dielectric triangles (surface current method) differ only slightly.

```
                    DATA OF THE DIELECTRIC TRIANGLES

no.       label     x1 in m     y1 in m     z1 in m       edges
         medium     x2 in m     y2 in m     z2 in m
         medium     x3 in m     y3 in m     z3 in m
                      nx          ny          nz          area in m*m
       1      0  -1.0000E-01  2.0000E-01 -8.5000E-01          1       2       3
              1  -1.0000E-01  4.0000E-01 -8.5000E-01
              0   1.0000E-01  2.0000E-01 -8.5000E-01
                  0.0000E+00  0.0000E+00 -1.0000E+00      2.0000E-02
       2      0   1.0000E-01  4.0000E-01 -8.5000E-01         -1       4       5
              1   1.0000E-01  2.0000E-01 -8.5000E-01
              0  -1.0000E-01  4.0000E-01 -8.5000E-01
                  0.0000E+00  0.0000E+00 -1.0000E+00      2.0000E-02
```

In this case an additional line gives the components (nx, ny, nz) of the normal vector of each triangle.

For the edges the extract is

```
                    DATA OF THE DIELECTRIC EDGES (with MoM)

                                                 triangle no.   points of the triangle ...
 no.  type      length/m        media     KORP   KORM  POIP  POIM  POIA  POIE
    1   3    2.8284E-01        0      1      1      2     1     1     3     2
    2   3    2.0000E-01        0      1      1      3     2     3     1     3
    3   3    2.0000E-01        0      1      1     26     3     3     1     2
                    electr. info of symmetry        magnet. info of symmetry
                    yz     xz     xy  status       yz     xz     xy  status
                     0      0     42  unknown       0      0     42  unknown
                     0      0     43  unknown       0      0     43  unknown
                     0      0     44  unknown       0      0     44  unknown
```

In addition to the data that is given for the metallic triangles, the following columns are provided `POIA`, `POIE`, `KNP` and `KNM`. The column `POIA` contains the number of the corner point of the triangle in `KORP`, where the basis function for magnetic surface current begins and the column `POIM` contains the number of the end point of the triangle where the basis function ends. The sizes `KNP` and `KNM` are the lengths when the vertices are connected to the middle of the opposite edge in the triangles `KORP` and `KORM`. The symmetry information is shown for the basis functions of both the equivalent electric or magnetic current densities.

The data for the segments follows the data for the triangles.

```
                          DATA OF THE SEGMENTS

 No.       label     x1 in m     y1 in m     z1 in m      nodes
           medium    x2 in m     y2 in m     z2 in m    length in m  radius in m
       1       0   0.0000E+00  0.0000E+00 -2.7700E+00         1
               0   0.0000E+00  0.0000E+00 -2.1625E-00   6.0750E-01   2.7700E-02
       2       0   0.0000E+00  0.0000E+00 -2.1625E+00        -1        2
               0   0.0000E+00  0.0000E+00 -1.5550E+00   6.0750E-01   2.7700E-02
       3       0   0.0000E+00  0.0000E+00 -1.5550E+00        -2        3
               0   0.0000E+00  0.0000E+00 -9.4750E-01   6.0750E-01   2.7700E-02
```

Here each segment is assigned a consecutive number. In the second column the label of the segment appears and below it the number of the medium in which it finds itself. A zero (0) means free space (vacuum). Then the coordinates of the begin and end points of the segment follow. In the previous row the numbers of the nodes, that are adjacent, appear. A positive sign for the node number indicates that the positive current direction is defined away from the segment. When there is a negative number then the positive direction is towards the segment.In the next row the length of the segment appears, followed by the radius.

For the data of the nodes between the segments a data table is given.

DATA OF THE NODES BETWEEN THE SEGMENTS

	no. of segment		points of segm.		...
No.	ISEGP	ISEGM	KNOP	KNOM	
1	1	2	2	1	
2	2	3	2	1	
3	3	4	2	1	

info of symmetry			
yz	xz	xy	status
0	0	5	unknown
0	0	6	unknown
0	0	7	unknown

The consecutive numbers of nodes are given in the first column. Then the number ISEGP and ISEGM of the two connected segments follow. A positive current direction is defined from the segment ISEGP to the segment ISEGM. The column KNOP indicates whether the begin point(KNOP = 1) of the segment ISEGP connects to the node or if it is the end point(KNOP = 2). The following four columns contain the data about the symmetry and are the same as for the metallic triangles described above.

If there are any connections between triangles and segment, then the following data is given.

GEOMETRIC DATA OF CONNECTIONS SEGMENTS - TRIANGLES

	Data of triang.		data of segm.			info of symmetry			
no.	DRENUM	DREPOI	SEGNUM	SEGPOI	angle	yz	xz	xy	status
1			11	1	360.0000	0	0	0	unknown
	15	1			45.0000				
	33	1			45.0000				
	55	1			45.0000				

Each connection point is assigned a consecutive number which is given in the first column. The number of the triangle at the connection point is given in the column DRENUM with the connecting vertex (1 to 3) in the column DREPOI. Likewise the connecting segment's number is given in the column SEGNUM and the connecting end in the SEGPOI column. (If the begin point of the segment is connected, SEGPOI=1; else the end point is connected and SEGPOI=2.) The column angle gives the angle that is formed by the triangle at the connection point (in radians). The meaning of the symmetry information in the last four columns is the same as that of the metallic triangles given above.

If a dielectric volume element has been used, then the following data block is given:

```
                    DATA OF THE DIELECTRIC CUBOIDS

   No.       x1 in m      y1 in m      z1 in m  rel.permittivity
   label     x2 in m      y2 in m      z2 in m  conductivity in S/m
             x3 in m      y3 in m      z3 in m  mass density in kg/m*m*m
             x4 in m      y4 in m      z4 in m  loss factor tan(delta))
       1  0.0000E+00   0.0000E+00   0.0000E+00    4.0000E+00
       0  3.3333E-01   0.0000E+00   0.0000E+00    0.0000E+00
          0.0000E+00   3.3333E-01   0.0000E+00    1.0000E+03
          0.0000E+00   0.0000E+00   3.3333E-01    0.0000E+00
       2  0.0000E+00   0.0000E+00   3.3333E-01    4.0000E+00
       0  3.3333E-01   0.0000E+00   3.3333E-01    0.0000E+00
          0.0000E+00   3.3333E-01   3.3333E-01    1.0000E+03
          0.0000E+00   0.0000E+00   6.6667E-01    0.0000E+00
```

Each cuboid is given a consecutive number. The x, y and z corner points coordinates are given in the first three columns. The first row is the reference point. The second row is the corner point, to which from the reference point the first basis function is defined. Further, the third and fourth rows define the next two basis functions with respect to the reference point. In the last column the relative dielectric constant ε_r followed by the conductivity σ in $\frac{1}{\Omega \mathrm{m}}$ as well as the density of the cuboid in $\frac{\mathrm{kg}}{\mathrm{m}^3}$.

In each dielectric cuboid there are three basis functions, in each coordinate direction one. The data of these basis functions is given in the following format:

```
          DATA OF THE BASIS FUNCTIONS FOR DIELECTRIC CUBOIDS

                              Symmetry information
 No. cuboidno. direc.    yz      xz      xy  status
     1        1  1        28      55     109  unknown
     2        2  1        29      56     110  unknown
     3        3  1        30      57     111  unknown
     4        4  1        31      58     112  unknown
```

In the first column the consecutive number of the basis function is given. The next column indicates the number of the cuboid. The column direction indicates the direction of the basis function in the respective cuboid. 1 indicates that e.g. the basis function is defined from the reference point to the second corner point. The last four columns contain information concerning the symmetry properties of the cuboid, where the structure and the meaning is the same as with the other basis functions.

Thereafter information follows regarding the number of basis functions.

```
                        DATA FOR MEMORY USAGE

 Number of metallic triangles:      0               max. triangles:MAXNDR   =    0
 Number of dielectric triangles:    0
 Number of metallic segments:       0               max. segments: MAXNSEG  =    0
 Number dielectr./magnet. cuboids: 64               max. cuboids:  MAXNQUA  =   64

 Number of metallic edges (MoM):    0  unknown:   0  max. edges:   MAXNKA    =   10
 Number of metallic edges (PO):     0  unknown:   0
 Number of dielectric edges (MoM):  0  unknown:   0
 Number of nodes between segments:  0  unknown:   7  max. nodes:   MAXNKNO   =   10
 Number of connection points:       0  unknown:   0  max. conn.:   MAXNV     =   15
 Number of dielectric cuboids:     64  unknown:   0  max. cuboids: MAXNQUA   =   64

 Number of basis funct. for MoM:  192  unknown:  48  max. basisf.  MAXNZEILE =  227
 Number of basis funct. for PO:     0  unknown:   0  max. basisf.  MAXNKAPO  =    0

 Memory requirement for the matrix:  48 rows *  192 columns =  9216 complex numbers
 For the matrix, a memory of NMAT = 9216 complex numbers is available
 Storing the matrix and solving the linear set of equations in main memory

     A total of 171780 bytes of memory have been allocated dynamically
```

Here the data, e.g. the number of basis functions on the nodes between segments, can be extracted. It is also indicated how many have the status `unknown`, i.e. how many have to be determined by solving the matrix equation. The maximum number of nodes is also given and this number may not be exceeded.

14.2 Excitation

The data here is structured depending on the means of excitation. For a voltage source on a segment the following data block is generated:

```
                EXCITATION BY VOLTAGE SOURCE AT SEGMENT

 Number of voltage source:    N =   1
 Frequency in Hz:             FREQ =  7.50000E+07
 Wavelength in m:             LAMBDA =  3.99723E+00
 Open circuit voltage in V:   |U0| =  1.00000E+00
 Phase in deg.:               ARG(U0) =     0.00
 Source at segment w. label:  ULA =     1
 Absolute number of segment:  UNR =    11
```

If an incident plane wave is used then the output file has the following format:

```
      EXCITATION BY PLANE LINEAR POLARISED ELECTROMAGNETIC WAVE

 Number of excitation:   N =   1
 Frequency in Hz:        FREQ = 1.9467E+07
 Wavelength in m:        LAMBDA = 1.5400E+01
 Direction of incidence: THETA = -180.00  PHI =     .00
 Dir. of polarisation:   ETA =    0.00
 Direction of propag.:   BETA0X = -1.6736E-10
                         BETA0Y =  0.0000E+00
                         BETA0Z =  4.0800E-01
 Field strength in V/m:  |E0X| =  1.00000E+00   ARG(E0X) =     0.00
 (Phase in deg.)         |E0Y| =  0.00000E+00   ARG(E0Y) =     0.00
                         |E0Z| =   4.1021E-10   ARG(E0Z) =     0.00
```

The vector $\vec{\beta}$, whose components are given, is the vector which points in the direction of propagation. The vector $\vec{E}_0$ represents the direction of the electric field.

14.3 Currents and charges

After solving the matrix equation, the currents are extracted, if extraction has not been suppressed by the parameter PS4 in the PS card.

A surface current, that flows perpendicularly to the edge, belongs to each edge. The coefficients appear in the table:

```
                  SURFACE CURRENT DENSITY ON TRIANGLES (in A/m)

no. node  real part  imag. part   magn.        phase
    1   -6.9254E-04  1.0356E-03  1.2458E-03   123.77
    2    0.0000E+00  0.0000E+00  0.0000E+00      .00
    3    1.0281E-03 -1.4961E-03  1.8153E-03   -55.50
    4   -1.0036E-03  1.3894E-03  1.7140E-03   125.84
    5    7.0292E-05  1.7210E-05  7.2369E-05    13.76
```

When examining segments, each node is assigned a current:

```
                       CURRENTS IN SEGMENTS (in A)

no. node  real part  imag. part   magn.        phase
    1    3.3953E-03 -1.7374E-03  3.8140E-03   -27.10
    2    6.2056E-03 -3.0587E-03  6.9185E-03   -26.24
    3    8.4702E-03 -3.9922E-03  9.3638E-03   -25.24
```

For connection points between segments and triangles, the current of each basis function is given:

```
                    CURRENTS AT CONNECTION POINTS (in A)

number    real part  imag. part   magn.        phase
   1   -2.6923E-03  9.6710E-04  2.8607E-03   160.24
   2   -2.6923E-03  9.6710E-04  2.8607E-03   160.24
   3   -2.6923E-03  9.6710E-04  2.8607E-03   160.24
```

For a dielectric solid (surface current method) the equivalent surface current

```
          EQUIVALTENT ELECTRIC CURRENTS ON TRIANGLES (in A/m)

no.of edge real part  imag. part   magn.        phase
    1   1.3380E-03 -7.8979E-04  1.5537E-03   -30.55
    2   0.0000E+00  0.0000E+00  0.0000E+00      .00
    3  -1.9138E-03  1.3635E-04  1.9187E-03   175.92
    4   1.5428E-03 -1.2464E-03  1.9834E-03   -38.93
    5  -2.8018E-04 -3.9938E-04  4.8786E-04  -125.05
```

and the equivalent magnetic current

```
          EQUIVALENT MAGNETIC CURRENTS ON  TRIANGLES (in V/m)

no. edge      real part  imag. part   magn.        phase
    1   1.0505E+00  4.9770E-01  1.1625E+00    25.35
    2   0.0000E+00  0.0000E+00  0.0000E+00      .00
    3   2.5042E-02 -5.8154E-01  5.8208E-01   -87.53
    4   9.1269E-01 -1.2119E-01  9.2071E-01    -7.56
    5  -9.2055E-01 -5.5293E-01  1.0738E+00  -149.01
```

is given for each basis function on an edge . In the case of dielectric cuboids the polarisation current is given for each cuboid:

```
            EQUIVALENT ELECTRIC CURRENTS AT VOLUME ELEMENTS (in A/m*m)

Cuboid-         I1                  I2                  I3
number    magn.     phase     magn.     phase     magn.     phase
    1  6.1821E-11  113.21  4.0629E-11 -127.75  4.4212E+00   52.33
    2  4.2163E-11  106.10  1.9748E-11 -127.90  3.7752E+00   52.58
    3  5.9327E-11   99.04  1.6515E-11 -128.34  2.4592E+00   53.12
```

The OS card can request the current distribution. Here the following data is given for each triangle

```
          VALUES OF THE CURRENT DENSITY VECTOR ON TRIANGLES in A/m

Triangle     centre                 JX              JY             JZ             ...
number    x/m    y/m    z/m      magn.    phase     magn.   phase    magn.    phase
 1      -.944   .000   .056   1.644E-03    19.10  0.000E+00    .00  3.716E-02 162.73
 2      -.889   .000   .111   1.184E-03   163.01  0.000E+00    .00  3.238E-02 157.26
 3      -.944   .000   .222   4.709E-03    12.49  0.000E+00    .00  2.784E-02 149.11
 4      -.889   .000   .278   2.032E-03  -170.73  0.000E+00    .00  2.081E-02 119.69
 5      -.944   .000   .389   4.285E-03    13.23  0.000E+00    .00  2.083E-02 100.40

                                                  ...    Current magnitude in the
                                                              3 corner points
                                                      3.850E-02 3.882E-02 3.457E-02
                                                      3.145E-02 3.169E-02 3.445E-02
                                                      3.143E-02 3.446E-02 2.197E-02
                                                      2.109E-02 2.234E-02 2.146E-02
                                                      2.106E-02 2.467E-02 2.291E-02
```

At the position (`x, y, z`) the current density vector $\vec{J}$ in the complex form is given. The last three columns indicate the value for the surface current density in the three vertices of the triangles, where the value is the average of the current at the vertices of all three adjacent triangles. If the current is requested, the charge on each triangle is also written to the output file. Only the charge is given as the position of each triangle is the same as written for the currents.

```
          VALUES OF THE SURFACE CHARGE DENSITY ON TRIANGLES in As/m^2

Triangle          SIGMA
number          magn.       phase
        1    2.50469E-13    56.08
        2    3.55072E-13    42.60
        3    9.33040E-13    54.20
```

The current on the segments is written as

```
                    VALUES OF THE CURRENT IN THE SEGMENTS in A

Segment         centre                                          IX          ...
number            x/m          y/m          z/m       magn.     phase
        1   2.25000E+00  0.00000E+00  6.67500E-01 0.000E+00     0.00
        2   2.25000E+00  0.00000E+00  5.02500E-01 0.000E+00     0.00
        3   2.25000E+00  0.00000E+00  3.37500E-01 0.000E+00     0.00
                                      IY                 IZ
                                   magn.     phase    magn.     phase
                                 0.000E+00     0.00 2.208E-03   145.51
                                 0.000E+00     0.00 6.118E-03   146.21
                                 0.000E+00     0.00 9.000E-03   147.38
```

With the associated charge

```
          VALUES OF THE LINE CHARGE DENSITY ON SEGMENTS in As/m

Segment               Q
number          magn.       phase
        1    4.26233E-11    55.51
        2    3.28910E-11    58.02
        3    2.28537E-11    62.48
```

For every voltage source the current at the feed point is determined and thus the impedance. The following is the result:

```
                        DATA OF THE VOLTAGE SOURCE NO.   1

                    real part imag. part      magn.          phase
 Current   in A    1.0888E-02 -4.4405E-03  1.1759E-02       -22.19
 Admitt.   in A/V  1.0888E-02 -4.4405E-03  1.1759E-02       -22.19
 Impedance in Ohm  7.8747E+01  3.2116E+01  8.5044E+01        22.19

                                Power in Watt: 5.44395E-03
```

14.4 Finite conductivity

Firstly the block with the set of characteristics for the single labels is displayed:

```
                          DATA OF LABELS

Label     2: DOSKIN =  3             DOLAST =  0                  ...
          Triangle thickness:  5.00000E-03 m
          Sigma =   1.000E+05 S/m   Mue_r =   1.000E+00
          Penetration depth of the skin effect:  1.59210E-04 m

All segments and triangles without a listed label are perfectly
                               DOCOVR =  0

                            tan(delta_mu) =   0.000E+00

                                  conducting
```

After the calculation of the currents the losses that result from finite conductivity are displayed.

```
                               POWER LOSS (in Watts)

     |               ...in the segments                 |   ...in the triangles
Label| skineffect  ohm.loss    distr.load   coating     | skineffect  ohm. loss
    2| 0.0000E+00  0.0000E+00  0.0000E+00   0.0000E+00  | 5.4620E-07  1.7386E-08
total| 0.0000E+00  0.0000E+00  0.0000E+00   0.0000E+00  | 5.4620E-07  1.7386E-08

          Total loss in the segments:         0.0000E+00 W
          Total loss in the triangl.:         5.6359E-07 W
          Loss (total)         :              5.6359E-07 W
          Efficiency of the ant.  :              99.9877 %
```

In the first column the label is displayed, the lowest row displays the sum.

14.5 Near field

If the near field is calculated, the following data is displayed:

VALUES OF THE ELECTRIC FIELD STRENGTH in V/m

(total field, incident and scattered)

	LOCATION			EX		EY		...
medium	X/m	Y/m	Z/m	magn.	phase	magn.	phase	
0	0.00000E+00	0.00000E+00	-1.00000E+00	6.70088E+01	99.86	7.65636E-01	166.42	
0	1.00000E-01	0.00000E+00	-1.00000E+00	6.46235E+01	74.23	1.14589E+00	166.13	
0	2.00000E-01	0.00000E+00	-1.00000E+00	6.23014E+01	47.98	1.55289E+00	165.83	
0	3.00000E-01	0.00000E+00	-1.00000E+00	5.99908E+01	21.51	1.95743E+00	163.95	

...	EZ	
	magn.	phase
	6.89061E+01	126.74
	7.17685E+01	98.76
	7.35678E+01	70.70
	7.41473E+01	42.30

Displayed are the position as well as the individual components of the electric and the magnetic field strength. This is the total value of the field, i.e. the sum of the incident wave and the scattered field.

If the electric field inside dielectric cuboids is determined, then the value for the SAR (*specific absorption rate*) and the cuboid number are also given:

VALUES OF THE ELECTRIC FIELD STRENGTH in V/m

inside the dielectric cuboids

LOCATION			EX		EY		EZ		...
X/m	Y/m	Z/m	magn.	phase	magn.	phase	magn.	phase	
0.050	0.050	0.050	5.776E+00	59.89	1.259E+01	-177.82	1.415E+02	-125.12	
0.050	0.050	0.150	2.192E+01	33.75	4.114E+00	122.93	1.640E+02	-130.45	
0.050	0.050	0.250	2.584E+01	31.18	3.420E+00	19.21	1.679E+02	-137.51	
0.050	0.050	0.350	2.625E+01	22.29	8.499E+00	-24.72	1.551E+02	-144.87	

...	SAR	cuboid no.
	in W/kg	
	0.000E+00	1
	0.000E+00	2
	0.000E+00	3
	0.000E+00	4

14.6 Far fields

If the far field is calculated, the following block in this form is displayed:

```
    VALUES OF THE SCATTERED ELECTRIC FIELD STRENGTH IN THE FAR FIELD in V
                  Factor e^(-j*BETA*R)/R not considered

   LOCATION          ETHETA             EPHI            directivity in dB   ...
 THETA   PHI      magn.     phase     magn.    phase     vert.     horiz.   total
 90.00   0.00   1.235E+00   168.98  0.000E+00   0.00     7.1722 -999.9999   7.1722
 90.00   2.00   1.233E+00   168.90  0.000E+00   0.00     7.1583 -999.9999   7.1583
 90.00   4.00   1.227E+00   168.65  0.000E+00   0.00     7.1166 -999.9999   7.1166
                                              POLARISATION
                                        axial r. angle   direction
                                        0.0000   180.00   LINEAR
                                        0.0000   180.00   LINEAR
                                        0.0000   180.00   LINEAR

    Gain is a factor of  1.00000E+00 (    0.00 dB) larger than directivity

      The directivity/gain is based on an active power of  4.88015E-03 W
      and on a power loss of  0.00000E+00 W
```

The values that are displayed here are the values of the scattered field, i.e. the incident field is not taken into account. The ϑ and φ components of $\vec{E}_{\text{far}}$ are tabulated. Here we have

$$\lim_{R\to\infty} \vec{E}(\vec{r}) \;=\; \frac{e^{-j\beta_0 R}}{R}\,\vec{E}_{\text{far}}$$

and $R = |\vec{r}|$. Please note that the dimension of $\vec{E}_{\text{far}}$ is voltage. If the excitation is an incident wave, the results include the radar cross section. In the case of voltage sources, the gain or directivity are included (see the parameters of the FF card). The gain/directivity output is split into vertical (ϑ) and horizontal (φ) components.

The last three columns give the polarisation information of the scattered wave. In general the polarisation is elliptical as shown in figure 14-1. The coordinates are $\vec{e}_r$, $\vec{e}_\vartheta$ and $\vec{e}_\varphi$, and the view is in the direction of the propagation of the wave ($\vec{e}_r$).

In the column `axial` the ratio between $\frac{E_{min}}{E_{max}}$ of the major and minor axes is given. A ratio of 0 means that the wave is a linearly polarised wave, but if the ratio has a value of 1 then it is a circularly polarised wave. In the column `angle` the angle α is given. It is the angle between the major axis and the unit vector $\vec{e}_\vartheta$. The last column gives the direction of the polarisation (left/right/linear).

If the FF card is used with `NTHETA` ≥ 2 and `NPHI` ≥ 2 the Pointing vector is integrated over the specified sector, see the detailed discussion given with the FF card in section 9.2.25. The result is the radiated power and is given below the field values.

When analysing an antenna the source power (calculated from the input impedance) should equal the integral of the radiated power over the surface of a sphere. This may be used as a partial validation of the result. Note that power losses in dielectrics and finite conductivity should be taken into account separately.

The use may also elect to integrate the far field power without writing the field values to the output file (using the FF card with FFREQ = 3). FEKO then produces the output

```
    VALUES OF THE SCATTERED ELECTRIC FIELD STRENGTH IN THE FAR FIELD in V
                Factor e^(-j*BETA*R)/R not considered

Integration of the normal component of the Poynting vector in the angular
grid DTHETA =    5.00 deg. and DPHI =    5.00 deg. (    2701 sample points)
  angular range THETA        angular range PHI       radiated power
 -2.50 .. 182.50 deg.      -2.50 .. 362.50 deg.     1.05642E-03 Watt
  0.00 .. 180.00 deg.       0.00 .. 360.00 deg.     1.04095E-03 Watt
Polarisation dependent radiated power:
     horizontal polarisation:    3.63158E-04 Watt ( 34.89 %)
     vertical polarisation:      6.77792E-04 Watt ( 65.11 %)
     S polarisation:             5.19105E-04 Watt ( 49.87 %)
     Z polarisation:             5.21845E-04 Watt ( 50.13 %)
     left hand circular pol.:    7.13564E-04 Watt ( 68.55 %)
     right hand circular pol.:   3.27386E-04 Watt ( 31.45 %)
```

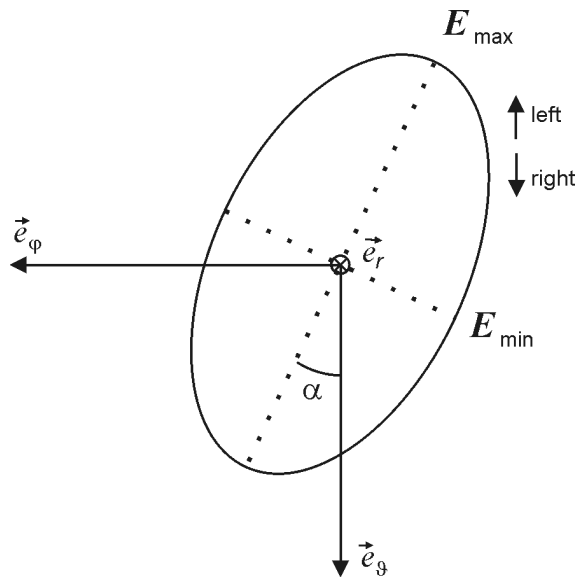


Figure 14-1: Elliptic polarisation in the far field

where the first line of total power is calculated assuming that each specified point lies at the centre of an incremental integration area. The effective area is therefore slightly larger than the area defined by the start and end angles. The second line gives the power integrated over an area defined by the start and end angles. For example, one may request an integration from $\varphi = 0°$ to $\varphi = 350°$ and $\vartheta = 5°$ to $\vartheta = 175°$ both in 10° increments in which case the first total will give the total power through the sphere. One may also request the integration from $\varphi = 0°$ to $\varphi = 360°$ and $\vartheta = 0°$ to $\vartheta = 180°$ in which case the second total will give the correct total power through the sphere.

The polarisation dependent power on the second block is calculated according to the effective area of the second line. The definitions of S and Z polarisation can be found in the discussion of the far field plotting in section 5.3.2.8.

14.7 S-parameters

If S-parameters have been requested with an SP card, FEKO prints different tables to the output file. The first lists the impedance at each port (all sources that are active when the SP card is processed are considered as ports).

LOAD IMPEDANCES AT PORTS

port	impedance in Ohm
1	5.00000E+01
2	1.00000E+02
3	5.00000E+01

Then the S-parameters are listed for each source as shown below. Note that sources whose amplitude are set to exactly zero are only used as sink ports, i.e. they are not excited and no such block is created. All the ports are loaded and FEKO therefore also writes this information to the output file. The second data line below gives S_{21} or the coupling to port 2 when port 1 is excited. In the second block here under the first line gives S_{13} or the coupling into port 1 when port 3 is excited.

SCATTERING PARAMETERS

	ports				magnitude		phase
	sink	source	real part	imag. part	linear	in dB	in deg.
S	1	1	6.14622E-02	-3.53596E-01	3.58898E-01	-8.90	-80.14
S	2	1	3.61992E-03	5.42992E-03	6.52594E-03	-43.71	56.31
S	3	1	-1.46490E-03	1.73598E-02	1.74215E-02	-35.18	94.82

...

SCATTERING PARAMETERS

	ports				magnitude		phase
	sink	source	real part	imag. part	linear	in dB	in deg.
S	1	3	-1.31791E-03	1.74114E-02	1.74612E-02	-35.16	94.33
S	2	3	9.17744E-01	1.08299E-01	9.24112E-01	-0.69	6.73
S	3	3	3.49405E-01	-4.49374E-02	3.52282E-01	-9.06	-7.33

14.8 Computation time

The final section in the output file gives an overview of the computation, in seconds, time in a tabular format:

SUMMARY OF REQUIRED TIMES IN SECONDS

	CPU-time	runtime
Reading and constructing the geometry	0.031	0.031
Checking the geometry	0.000	0.000
Initialisation of the Greens function	0.000	0.000
Calcul. of coupling for PO/Fock	0.000	0.000
Calcul. of matrix A	0.031	0.031
Calcul. of vector Y (right side)	0.000	0.000
Solution of the linear set of eqns.	0.016	0.016
Determination of surface currents	0.000	0.000
Calcul. of losses	0.000	0.000
Calcul. of electric near field	0.000	0.000
Calcul. of magnetic near field	0.000	0.000
Calcul. of far field	0.031	0.031
other	0.000	0.000
total times:	0.109	0.109

Index

** card, 8-3, 9-4
2D plots
    far fields, 3-14
    near fields, 3-16
3D patterns, 3-15
    display, 3-24

A0 card, 9-8
A1 card, 9-11
A2 card, 9-12
A3 card, 9-13
A4 card, 9-14
A5 card, 9-16
A6 card, 9-17
A7 card, 9-19
AC card, 9-20
ADAPTFEKO, 13-1
    plotting in GraphFEKO, 5-16
adaptive
    frequency sampling, 13-1
add lines, in GraphFEKO, 5-19
advanced visibility, 3-28
AE card, 9-23
AI card, 9-25
animation
    currents, 3-21
    near fields, 3-20
anisotropic layers, 9-89
    display in WinFEKO, 3-25
antenna parameters, 3-5, 3-13, 5-6
    display in GraphFEKO, 5-2
AP card, 9-27
aperture, 9-27
AR card, 9-34
array sizes, 2-12, 2-13
AV card, 9-39
axial ratio
    plotting in GraphFEKO, 5-12
axis
    display, 3-23
    length setting of, 3-31

BL card, 8-4
BO card, 9-41
border
    curved PO edge, 8-14
    PO correction edge, 8-49
    PO correction wedge, 8-55
BP card, 8-6
BQ card, 8-8
BT card, 8-11

CableMod, 9-20, 9-46
CAD program
    configuring, 3-10, 3-32
capacitance
    loading, 9-75, 9-77, 9-78
card editing, 4-2
CB card, 8-13
CG card, 9-43
circular cone, 8-50
circular disc, 8-56
circular hole, 8-73
CL card, 8-14
clear
    model information, 3-11
    output data, 3-11, 3-23
close graph, 5-3
close project, 3-7
CM card, 9-46
CN card, 8-17
CO card, 9-47
coarse segmentation, 8-47
coating of wires, 9-47
coaxial attachment approximation, 9-14
coil, 8-29
command line
    parameters
        WinFEKO, 3-33
comments, 8-3, 9-4
cone, 8-50
configuring parallel Windows version, 3-12
connection points
    definition, 2-4
continuous frequency
    plotting in GraphFEKO, 5-16
contour plots
    near fields, 3-18

control cards, 2-1, 9-1
** , 9-4
A0, 9-8
A1, 9-11
A2, 9-12
A3, 9-13
A4, 9-14
A5, 9-16
A6, 9-17
A7, 9-19
AC, 9-20
AE, 9-23
AI, 9-25
AP, 9-27
AR, 9-34
AV, 9-39
BO, 9-41
CG, 9-43
CM, 9-46
CO, 9-47
DA, 9-50
DI, 9-52
EN, 9-53
FE, 9-54
FF, 9-63
FR, 9-65
GF, 9-67
L4, 9-74
LD, 9-75
LE, 9-76
LP, 9-77
LS, 9-78
LZ, 9-79
OF, 9-80
OS, 9-81
PS, 9-83
PW, 9-85
SK, 9-89
SP, 9-93
TL, 9-94
coordinates, selection, 3-4
copy
geometry, 8-92
project, 3-8
coupling, 9-93
transmission line, 9-20, 9-46
create project, 3-3
cuboids, 8-18, 8-86
definition, 2-4
display, 3-25
current sources
line segment, 9-20, 9-25, 9-39
currents
animation, 3-21
calculation request, 9-81
display, 3-5, 3-21, 3-24
in GraphFEKO, 5-2, 5-8, 5-10, 5-11
on segments, 3-14, 3-21, 5-11
on surface elements, 3-21
selection, 3-4
curved arc, 8-14
cutplane
display, 3-23
options, 3-4, 3-27
cylinder, 8-108
dielectric, 8-21
UTD region, 8-101

DA card, 9-50
DI card, 9-52
dielectric, 2-11, 8-63, 9-52
cuboid cylinder, 8-21
cuboids, 8-18, 8-86
sphere, 9-67
thin sheet, 9-89
diffraction theory, 8-99
dimension, scaling, 8-88
dipole
aperture array, 9-27
directivity
plotting in GraphFEKO, 5-12
disc, 8-56
discrete elements, 9-77–9-79
display
axis, 3-23
currents, 3-5
`*.fek` file, 3-4, 3-11
multilayer substrates, 3-26
`*.neu` file, 3-4, 3-10
options, 3-4
for `*.fek` file, 3-26
for `*.neu` file, 3-27
WinFEKO, 3-1, 3-23
requested field points, 3-4

distributed load, 9-75
DK card, 8-18
DP card, 6-9, 8-20
dynamic memory management, 2-12
DZ card, 8-21

edges
    definition, 2-4
edit graph, 5-16
EditFEKO, 4-1
    card editor, 4-2
    edit menu, 4-5
    file menu, 4-1
    keystrokes, 4-7
    OPTFEKO mode, 4-6
    options menu, 4-2
    parameter suggestion, 4-4
    running from WinFEKO, 3-4, 3-10
    searching, 4-6
    superuser mode, 4-2
    variables, 4-5
effective gain
    plotting in GraphFEKO, 5-12
efficiency, 9-85
EG card, 8-23
EL card, 8-26
electric fields
    calculating, 9-54
    plotting in GraphFEKO, 5-14
elements, 8-11
    creation, *see* geometry cards
    display, 3-25
    display direction, 3-23
    display numbers, 3-23
    search for, 3-29
ellipsoid, 8-26
elliptical hole, 8-73
ELSE statement, 6-8
EM properties
    display, 3-26
EN card, 9-53
end of geometry, 8-23
end of input file, 9-53
environment variables, 2-15, 3-2
equivalent aperture, 9-27
excitation, *see* sources, 9-5
    display, 3-24, 3-29
EXIT command, 6-9
export graph data, 5-4

far fields
    2D plots, 3-5, 3-14
    3D plots, 3-5
    3D polar plots, 3-15
    calculating, 9-63
    plotting, 5-2
    plotting in GraphFEKO, 5-12
fast rotation, 3-23
FE card, 9-54
feed, *see* sources
`*.fek` file
    display, 3-4, 3-11, 3-26
FEKO, 7-1
    running from WinFEKO, 3-4, 3-11
FEKO_USER_HOME
    environment variable, 2-15, 3-2
FEKO_WRITE
    environment variable, 2-15
FEMAP
    `*.mod` file creation, 3-6
    running from WinFEKO, 3-4, 3-10
FF card, 9-63
field selection, 3-4
files, 9-50
    input, 2-1
    output, 14-1
    summary of, 2-2
find
    element, 3-4, 3-29
    FEMAP element, 3-29
flat display
    triangulated surfaces, 3-25
FO card, 8-28
Fock area, 8-28
FOR/NEXT loops, 6-6
FR card, 9-65, 13-1
frequency, 9-65
    adaptive sampling, 13-1

gain
    plotting in GraphFEKO, 5-12
general comments, 2-1
general settings
    options, 3-31

geometry
display, 3-24
entering, 2-4, 8-31
geometry cards, 2-1, 8-1
**, 8-3
BL, 8-4
BP, 8-6
BQ, 8-8
BT, 8-11
CB, 8-13
CL, 8-14
CN, 8-17
DK, 8-18
DP, 8-20
DZ, 8-21
EG, 8-23
EL, 8-26
FO, 8-28
HE, 8-29
IN, 8-31
IP, 8-47
KA, 8-49
KK, 8-50
KL, 8-55
KR, 8-56
KU, 8-60
LA, 8-62
ME, 8-63
NU, 8-67
PB, 8-71
PH, 8-73
PM, 8-78
PO, 8-81
PY, 8-84
QU, 8-86
SF, 8-88
SU, 8-90
SY, 8-91
TG, 8-92
TO, 8-95
TP, 8-98
UT, 8-99
UZ, 8-101
VS, 8-103
WG, 8-106
ZY, 8-108
GF card, 9-67
GraphFEKO, 5-1
clear file data, 5-6
data extraction, 5-2
edit menu, 5-16
file control, 5-1
file menu, 5-2
help, 5-21
import menu, 5-5
line arithmetics, 5-19
load/import data, 5-4
running, 3-5, 5-1
save/export data, 5-4
templates, 5-3
toolbars, 5-1
tools menu, 5-19
version information, 5-21
window menu, 5-20
graphics cards with WinFEKO, 3-2
graphs, arithmetics, 5-19, 5-20
Green's functions, 9-67
ground plane, 9-41
display, 3-24

hardware rendering, 3-2
HE card, 8-29
helix, 8-29
help
WinFEKO, 3-33
Hertzian electric dipole, 9-16
hide structure using cutplane, 3-27
hot-keys
EditFEKO, 4-7
WinFEKO, 3-2, 3-23

IF statement, 6-8
imaginary, in GraphFEKO, 5-19
impedance, 9-79
loading, 9-76
microstrip fed, 9-76
Smith chart, 5-7
import data in GraphFEKO, 5-4
import geometrical data, 8-31
impressed line current, 9-20, 9-25, 9-39
display shrinked cells, 3-28
enlarge radius, 3-28
IN card, 8-31
incident plane wave, 9-8

include files, 8-31
inductance
    loading, 9-75, 9-77, 9-78
input file, 2-1
IP card, 8-47
iso-surfaces
    near fields, 3-5, 3-16
isometric view, 3-5

KA card, 8-49
KK card, 8-50
KL card, 8-55
KR card, 8-56
KU card, 8-60

L4 card, 9-74
LA card, 8-62
label selected calculation, 9-80
labels, 8-13, 8-62
    display, 3-4, 3-26
layer display, 3-27
LD card, 9-75
LE card, 9-76
lead lines display, 3-23
legend display, 3-23
LFFEKO, 12-1
line segments, *see* segments
linear set of equations, 9-43
load
    subtract in GraphFEKO, 5-6, 5-16
load data
    in GraphFEKO, 5-1, 5-4
    in WinFEKO, 3-5
loading, 9-94
    an edge, 9-76
    attachment point, 9-74
    distributed, 9-75
    impedance, 9-79
    microstrip line, 9-76
    parallel circuit, 9-77
    series circuit, 9-78
log, in GraphFEKO, 5-20
losses, 9-85
low frequencies, 12-1
LP card, 9-77
LS card, 9-78
LZ card, 9-79

magnetic cuboids, 8-18
magnetic dipole, 9-17
magnetic fields
    calculating, 9-54
    plotting in GraphFEKO, 5-14
magnetic ring current, 9-13
magnitude, in GraphFEKO, 5-19
main display options, 3-4, 3-23
maxalloc(m), 2-12, 2-13
maximum constants, 2-12, 2-13
ME card, 8-63
medium
    dielectric, 8-63, 8-86
    magnetic, 8-86
memory
    allocation, 2-12
    releasing, 3-7, 3-11
    requirements for WinFEKO, 3-1
meshing, 2-4
    non-uniform, 8-6, 8-8, 8-11
    rules, 2-4
multilayer substrates
    defining, 9-67
    display, 3-26
multiple reflections, 8-81
multiply results in GraphFEKO, 5-19

near fields
    animation, 3-20
    calculating, 9-54
    calculation offset, 9-80
    contour plots, 3-18
    display, 3-5, 3-24
    iso-surfaces, 3-5, 3-16
    ortho-slices, 3-5, 3-18
    plotting, 3-16, 5-2, 5-14
    pointing vector, 3-16
network parameters, 3-5, 3-14, 5-8
    display in GraphFEKO, 5-2
networks, 9-94
`*.neu` file display, 3-4, 3-10, 3-27
new project, 3-6
nodes
    defining, 8-20
    definition, 2-4
    display, 3-23
    variable names, 6-9, 8-20

non-uniform mesh, 8-6, 8-8
normal vectors, 8-17
    display, 3-23
normalise model, 3-5
NU card, 8-67
NURBS surfaces, 8-67

OF card, 9-80
offset for near field calculation, 9-80
ohmic losses, 9-89
online help, 3-33, 5-21
open graph, 5-1, 5-2
open project, 3-3, 3-7
    list of previous projects, 3-9
OpenGL, 3-1
    hardware test, 3-30
    performance, 3-30
OPTFEKO, 10-1
    editing files, 4-6
options
    EditFEKO, 4-2
    WinFEKO
        general settings, 3-31
        toolbars, 3-33
origin of rotation, 3-30
ortho-slices
    near fields, 3-5, 3-18
    set transparency, 3-18
OS card, 9-81
out-of-core, 2-12
output file, 14-1
    viewing, 3-5, 3-13

pan
    detail settings, 3-30
    speed buttons, 3-5
paraboloid, 8-71
parallel version
    running in Windows, 3-12
parallelogram, 8-6
parameters of segmentation, 8-47
PB card, 8-71
PH card, 8-73
phase, in GraphFEKO, 5-19
phase, unwrapping, 5-20
physical optics, 8-81
picking, 3-30
planar substrate, 9-67
plane wave incidence, 9-8
plate with hole, 8-73
PM card, 8-78
PO border
    curved, 8-14
    edge, 8-49
    wedge, 8-55
PO card, 8-81
PO visibility, 8-103
point names, *see* nodes
polygons, 8-84
    definition, 2-4
    display, 3-25
    meshed, 8-78
    UTD formulation, 8-99
power density
    plotting in GraphFEKO, 5-14
power input, 9-85
`*.pre` files
    opening with WinFEKO, 3-7
PREFEKO, 6-1
    running from WinFEKO, 3-4, 3-11
previous display, 3-26
    options, 3-4
    return to, 3-5
PRINT command, 6-9
printing, 3-8
    in GraphFEKO, 5-1, 5-5
    in WinFEKO, 3-3
    metafile, 3-9
    PostScript, 3-9, 5-5
    printer setup, 3-8
    to clipboard, 3-8
    vector based, 3-9
priority, setting, 7-1
program configuration
    loading, 3-31
    saving, 3-31
program execution control, 9-83
project management, 3-3
PS card, 9-83
PW card, 9-85
PY card, 8-84

QU card, 8-86
quadrangle, 8-8

radiation
input file templates, 3-6
patterns, 9-63
patterns as sources, 9-34
ray paths in WinFEKO, 3-22
real ground, 9-41, 9-67
real part
in GraphFEKO, 5-19
receiving antenna, 3-5, 3-14, 5-10
display in GraphFEKO, 5-2
requested field display, 3-24, 3-28
resistance loading, 9-75–9-78
results, 3-5
checking validity, 2-18
extracting from output file, 3-13
rotation, 8-92
around arbitrary selected point, 3-30
of point, 8-98
WinFEKO speed buttons, 3-5

S-parameters, 3-14, 5-8, 9-93
S_{11}, 3-13, 3-14, 5-6
S_{21}, 3-14, 5-8
SAR
calculations in WinFEKO, 3-16
plotting in GraphFEKO, 5-14
save
as UNIX, 4-1
graph, 5-1, 5-2
graph data, 5-1, 5-4
project, 3-3, 3-7
WinFEKO options, 3-31
scaling, 8-88, 8-92
scattering
input file templates, 3-6
plotting in GraphFEKO, 5-12
searching for element, 3-4
segmentation
parameters, 8-47
rules for, 2-4
segments, 8-4
arc, 8-14
coating, 9-47
creation, *see* geometry cards
definition, 2-4
display, 3-25, 3-28
helix, 8-29
selecting (picking)
geometry, 3-30
results, 3-30
selection
far field data, 3-15
geometry and results, 3-4
of current data, 3-21
of near field data, 3-18
SF card, 8-88
SK card, 9-89
skin effect, 9-89
Smith chart, 5-6, 5-7
smooth surface display, 3-25
solution information, 3-23
sources, 9-5
Hertzian electric dipole, 9-16
impressed line current, 9-20, 9-25, 9-39
incident plane wave, 9-8
magnetic dipole, 9-17
magnetic ring current, 9-13
microstrip line, 9-23
patch feed pin, 9-14
radiation patterns, 9-34
voltage on a node, 9-12
voltage on a segment, 9-11
voltage on an edge, 9-19, 9-23
SP card, 9-93
sphere, 8-60
dielectric/magnetic, 8-18
spiral, 8-29
square root, in GraphFEKO, 5-20
square, in GraphFEKO, 5-20
stripline
feeding, 9-23
loading, 9-74
SU card, 8-90
substrate, 9-67
subtract lines, in GraphFEKO, 5-19
superuser mode
in EditFEKO, 4-2
in FEKO, 8-90
surface currents
distribution, 9-81
with WinFEKO, 3-5
surface elements, *see* elements
surface triangles, *see* elements
SY card, 8-91

symbolic variables, 6-1
symmetry, 2-7, 8-91
    electric, 2-7
    geometric, 2-7
    magnetic, 2-8

templates, 3-6
    for GraphFEKO, 5-3
TG card, 8-92
thin dielectric sheet, 9-89
third party CAD
    running from WinFEKO, 3-10
    setting options for, 3-32
TIMEFEKO, 11-1
TL card, 9-94
TO card, 8-95
toolbars, visibility, 3-33
tools in WinFEKO, 3-29
    find element, 3-4
toroidal segment, 8-95
TP card, 8-98
transformation, 8-92
    model, 3-30
    of point, 8-98
translation, 8-92, 8-98
transmission line, 9-94
transmission line coupling, 9-20, 9-46
triangles, *see* elements

unwrap phase, 5-20
UT card, 8-99
UTD, 8-99
    cylinder, 8-101
    polygon, 8-84
UZ card, 8-101

variables
    environment, 2-15
    in point names, 6-9
    memory allocation, 2-12
    predefined, 2-12
    symbolic, 6-1
version information, 3-33
view
    angle settings, 3-30
    output file, 3-13
visibility
    PO region, 8-103
voltage source
    on a node, 9-12
    on a segment, 9-11
    on edge, 9-19, 9-23
VS card, 8-103

warnings
    reporting in WinFEKO, 3-32
wedge as PO border, 8-55
WG card, 8-106
WinFEKO, 3-1
    display excitation, 3-29
    display requested fields, 3-28
    display settings, 3-1, 3-23
    file control, 3-3
    file menu, 3-6
    graphics card, 3-2
    hardware requirements, 3-1
    hot-keys, 3-2
    ini file, 3-2
    loading results, 3-13
    memory requirements, 3-1
    preprocessing menu, 3-10
    running, 3-2
    solve menu, 3-11
    toolbars, 3-3
    tools, 3-29
    under MS Windows, 3-1
wire
    curved, 8-14
wire grid, 8-106
wire segments, *see* segments
write access restrictions, 3-32

Z_{in}, 3-13, 5-6
zoom
    detail settings, 3-30
    previous zoom, 3-5
    speed buttons, 3-5
    to window, 3-5
ZY card, 8-108